New Decomposition Methods for Economic Equilibrium Models with Applications to Decomposition by Region

by

William S. W. Chung

A thesis
presented to the University of Waterloo
in fulfilment of the
thesis requirement for the degree of
Doctor of Philosophy
in
Management Sciences

Waterloo. Ontario. Canada. 1999

0-612-44754-5

The University of Waterloo requires the signatures of all persons using or photocopying this thesis. Please sign below. and give address and date.

Abstract

Economic equilibrium models often distinguish several regions. For model management reasons. it can be advantageous to keep the submodels separate. even on different computers. e.g.. different personnel may be in charge of developing and maintaining the demand and supply submodels. Therefore. decomposition procedures can be useful to bring the submodels together for a solution.

Although existing decomposition principles can make large-scale linear or nonlinear programming models more manageable. economic equilibrium modelers cannot always use these techniques because many equilibrium models cannot be converted into optimization problems. This dissertation develops new decomposition methods by which existing LP decomposition principles can be applied to economic equilibrium models (non-optimization models). Preliminary tests are included.

Acknowledgements

Thank God (Jesus).

Thank God for giving me enough strength to go through the most difficult time when Donna (my wife) had two vital surgeries in the whole second year of my Ph.D. studies. In that year. comprehensive examination. course work for the comprehensive examination requirement. and Donna's surgeries almost drove me mad.

I would like to express my heartfelt gratitude to my supervisors. Dr. David Fuller and Dr. June Wu. for their enthusiasm. encouragement. and generous financial support. Especially. when Donna was having two vital surgeries. their patience and understanding became the invaluable support for me to resume my research work.

I would also like to thank Gladys and Christopher Wong who lent their laptop to me so that I could work for my research in Toronto's hospitals with Donna.

Thanks also go to the examination committee: Dr. Anna Nagurney. Dr. K. Ponnambalam. Dr. Trien Nguyen. and Dr. Ray G. Vickson. Their valuable comments have enriched the content of this thesis.

Finally. this work would not have been possible without the love. patience. and support of my wife. my parents. and my younger sister.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Economic equilibrium models (EEM) have been used to study energy-environment policies. tax policies and traffic policies for decades. An example of the EEMs is the energy equilibrium model which includes a detailed supply process model and demand functions. This dissertation is motivated by our several years of applied research with a multi-regional energy equilibrium model - a Canada-USA energy equilibrium model [8]. which consists of six regional energy equilibrium submodels. Regional submodels are linked with each other by a few energy flows. We noticed that if these six regional submodels could be developed. solved and debugged individually and then integrated together as a whole multi-regional model. the modeling task would be greatly reduced. Other research groups (e.g. [18]. [7]. and [3]) have similar observations. Hence. we chose to investigate methods to integrate the submodels of a multi-regional economic equilibrium model (MREEM). which is defined as an economic equilibrium model in which two or more regional economic equilibrium models are connected by a few linking economic activities among them. The linking activities may be flows of commodities among regions. or joint upper limits on emissions of pollutants. for example.

In general. it would be desirable to be able to develop a new MREEM efficiently by integrating all existing regional EEMs. created by one or many regional research groups. by introducing a few linking economic activities among them. An integration example can be found in the energy equilibrium models. Regional research groups often maintain their detailed energy equilibrium models. If all regional energy equilibrium models can be converted into optimization models (which may be unrealistic for multi-fuel models. see Chapter 2). existing decomposition algorithms of linear or nonlinear programming (LP or NLP) may be used as an integration means. However. realistic multicommodity energy equilibrium models cannot normally be represented as optimization models.

Obviously. if some or all of the regional EEMs are not optimization models. existing LP or NLP decomposition principles cannot be used. Hence. the main objective of this dissertation is to provide new methods by which existing LP decomposition principles can be adapted to decompose non-optimization EEMs. With these methods. we hope to alleviate the difficulties of model development and maintenance. which are due to the complexity and large-scale nature of multi-regional models.

Often. new algorithms are developed for the sake of reducing the computing time needed to solve a problem. However. solution time. compared with modeling time. is not an issue of concern for many large-scale equilibrium models. Murphy (1993) [21] provided a detailed discussion on how decomposition methods can be used to reduce the modelling time of large-scale models. He illustrated with an example showing that the modelling time can be reduced around 70% by means of decomposition methods. Besides the reduction of the modelling time. he also mentioned different advantages such as error reduction. Therefore. we view a decomposition method as successful if it finds an equilibrium reliably. without an unacceptably large increase in solution time.

Another motivation for decomposition is to overcome limits on model size due to computing capacity problems. Although each regional EEM may be solvable on a computer available to the modeler. the new MREEM may become too huge to be solved as a single model on one computer. Based on the decomposition-based information exchange provided by the new methods. modelers do not need to put the entire MREEM on one computer. Instead. the regional models can be solved on their own computers and they can exchange information with each other over a network. in order to solve the MREEM.

The new methods presented in this thesis show how modelers can integrate their regional EEMs as a new MREEM by directly adapting existing LP decomposition principles. The integrated MREEM is to be solved by exchanging the price and quantity information for the linking activities among all regional EEMs in an iterative manner until an equilibrium is found. The pattern of the decomposition-based information exchange follows that of the adapted decomposition principle. At each iteration. all regional EEMs can be solved by appropriate algorithms for finding equilibrium solutions.

Chapter 2 provides the formulation of the MREEM and explains when the

MREEM cannot be converted into an optimization model. The solution methods and the existing decomposition methods for the non-optimization MREEM are reviewed. The literature review shows that there is no appropriate decomposition method for the non-optimization MREEM.

Chapter 3 describes a new decomposition method consisting of three steps. An illustration of the new decomposition method is given to show how the Dantzig-Wolfe [9] decomposition principle is adapted by the new method. We also demonstrate how the new decomposition method can adapt another existing LP decomposition principle. The new method can be viewed as a general procedure for decomposing non-optimization equilibrium models. The formulation of the decomposed MREEM in variational inequalities (VI) form is presented to show that the new decomposition method can be used in that more general setting.

Chapter 4 shows that there is a computational difficulty (divergence) when we apply the new decomposition method in Chapter 3 to solve a MREEM. which includes the consideration of the time-lagged effect in the demand side. The demand side of the MREEM in Chapter 3 does not consider the time-lagged effect which can make the MREEM more realistic. For example. in energy equilibrium models. when dealing with the effects of a decrease in the price of electricity. durable commodities (e.g.. a gas stove) may have to be worn out before a shift is made to the product whose price has fallen. In order to resolve the computational difficulty. another new decomposition method. which adapts the Dantzig-Wolfe decomposition principle [9]. is developed.

In Chapter 5. a new demand-supply decomposition method. following the similar approach of the new decomposition method in Chapter 3 which adapts the Dantzig-Wolfe decomposition principle. is developed to decompose an economic equilibrium model into a demand submodel and several supply submodels. Because such a demand-supply decomposition method inherits the finite convergence property of the Dantzig-Wolfe principle. it must converge in a finite number of iterations.

Chapter 6 concludes with the summary of the work and recommendations for future research.

Chapter 2

Background and Literature Review

In this Chapter. a general background of economic equilibrium models is provided first. Because multi-regional economic equilibrium models are used to derive the new decomposition methods. the formulation of the models is then presented. The literature review of the solution techniques and the existing decomposition methods used in economic equilibrium models follows.

2.1 Economic Equilibrium Model (EEM)

This thesis is concerned with the development of new decomposition methods for solving economic equilibrium models. A general background of economic equilibrium models is therefore presented first. Since Ahn and Hogan [2] presented the background concisely. the content of this section mainly is quotation from them with a few notation changes for the consistent presentation of this thesis. Based on Ahn and Hogan (1982) [2], economic equilibrium models with linearized production activities can be formulated as follows.

> **EEM**
>
> Supply side:
>
> $$\min_{x} \quad c^T x$$
>
> $$s.t. \quad Ax \geq q \quad (v) \tag{2.1}$$
>
> $$x \in Z \tag{2.2}$$
>
> Demand side: $q = Q(p)$
>
> Equilibrium condition: $p = v$
>
> where v is an optimal dual variable vector (shadow price vector) corresponding to the demand constraints $Ax \geq q$. c is a cost vector for the supply activities. x is a production activity level vector. Z is the convex polyhedral production feasibility set which includes resource availability constraints. material balance equations and other system constraints. $Ax \geq q$ are demand requirement constraints. and $Q(\bullet)$ is a vector-valued demand function defined over prices. p.
>
> Under the competitive market assumption. the elements of a shadow price vector v can be regarded as the supply prices. representing a bound on the marginal cost of meeting an additional unit of demand. Then. a solution v^*(or p^*) and q^* become equilibrium price and quantity vectors. respectively. An accompanying solution x^* represents an equilibrium production profile of the supply activities.

After the description of the economic equilibrium model. Ahn and Hogan also provided the following general discussion regarding the properties of the EEM.

> Consider a state of a multi-commodity market characterized by commodity prices $p = (p_1, \ldots, p_n)$ and consumption levels $q = (q_1, \ldots, q_n)$. Suppose that the consumers' behavior in this market is captured by a demand function $Q(\bullet)$ and the supply side is described by an indirect supply mapping $P(\bullet)$. Let these supply and demand mappings satisfy the following:

Assumption 1. The demand function $Q(\bullet)$ from R^n_+ into R^n_+ is continuously differentiable. Furthermore. the own price effects on demand levels are negative. i.e.. $\partial Q_i(p)/\partial p_i < 0$ for any p in R^n_+ and each i.

Assumption 2. The supply mapping $P(\bullet)$ is a point-to-set mapping from R^n_+ into R^n_+. (A point-to-set mapping $P(\bullet)$ is monotone if. for any q^1 and q^2 in its domain and for any p^1 and p^2 such that $p^1 \in P(q^1)$ and $p^2 \in P(q^2)$. it follows that $(p^1 - p^2)^T(q^1 - q^2) \geq 0$. If the above inequality is strictly positive for any $q^1 \neq q^2$. the mapping is said to be strictly monotone. A point-to-set mapping $P(\bullet)$ is said to be upper semi-continuous at $q \in R^n_+$. if assumptions $q^k \rightarrow q$ and $p^k \rightarrow p$ such that $p^k \in P(q^k)$ imply that $p \in P(q)$. $P(\bullet)$ is said to be upper semi-continuous on X if it is upper semi-continuous at each point of $X \subset R^n_+$. Note that upper semicontinuity is a generalization of continuity of a point-to-point mapping.)

With these supply and demand sides. a typical economic equilibrium condition is defined as follows:

Definition 1. A supply price vector v^*. a demand price vector p^*. and a supply level vector q^* constitute an equilibrium of the EEM if

$$\left.\begin{array}{rcl} v^* & \in & P(q^*) \\ q^* & = & Q(p^*) \\ p^* & = & v^*. \end{array}\right\} \qquad (2.3)$$

or. equivalently.

$$\left.\begin{array}{rcl} p^* & \in & P(q^*) \\ q^* & = & Q(p^*). \end{array}\right\} \qquad (2.4)$$

This is the general formulation of market equilibrium with a given demand function and a supply mapping. If we go one step further. (2.4) can be put into the form

$$p^* \in P(Q(p^*)) \qquad (2.5)$$

which is a fixed-point problem.

The EEM can also be cast in the above form after we identify the implicit supply mapping embedded in the cost minimizing supply model. This conversion makes it possible to analyze the EEM in terms of (2.3). (2.4). or (2.5).

Let $V(\bullet)$ be the total cost function of the supply activities:

$$V(q) = min_x\{c^T x | x \in Z, \quad Ax \geq q\}. \tag{2.6}$$

This function is known to be convex. piecewise-linear. and sub-differentiable with sub-gradient

$$\partial V(q^1) = \{v | V(q) - V(q^1) \geq v^T(q - q^1) \forall q \in \text{ the domain of } V(\bullet)\}. \tag{2.7}$$

The resulting mapping $\partial V(\bullet)$ is said to be the sub-gradient mapping of $V(\bullet)$. The image set $\partial V(q)$ for a given q is closed and convex, and the mapping $\partial V(\bullet)$ is upper semi-continuous and monotone: see Rockafellar (1969) [29].

In addition. the optimal dual variable $v(q)$ corresponding to the demand requirement constraint $Ax \geq q$ in the cost minimizing LP supply side is an element of the sub-gradient $\partial V(q)$: i.e..

$$v(q) \in \partial V(q). \tag{2.8}$$

This implies that under the competitive market assumption we can interpret the sub-gradient as the set of supply prices. and that the sub-gradient mapping $\partial V(\bullet)$ can be viewed as the inverse supply mapping of the supply side of the EEM. i.e..

$$P(q) = \partial V(q) \qquad \forall q. \tag{2.9}$$

The formulation in terms of monotone mappings leads immediately to a uniqueness result:

Lemma 1. Assume that the supply mapping $P(\bullet)$ is monotone and the negative of the demand function $-Q(\bullet)$ is strictly monotone. Then there exists at most one solution to (2.3). (2.4). or (2.5).

Corollary 1. Assume that $-Q(\bullet)$ is strictly monotone. Then. the EEM has at most one equilibrium.

The proofs of Lemma 1 and Corollary 1 can be found in Ahn and Hogan (1982) [2].

2.2 Multi-Regional Economic Equilibrium Model (MREEM)

Assume that we have two regional EEM submodels. named Region-1 and Region-2. with the following formulation. The subscript is the region index.

Region-1

Supply side:

$$\begin{aligned} &\min_{x_1} \quad c_1^T x_1 \\ s.t. \quad & A_1 x_1 \geq q_1 \qquad (v_1) \\ & B_1 x_1 \leq b_1 \\ & x_1. q_1 \geq 0 \end{aligned}$$

Demand side: $q_1 = Q_1(p_1)$

Equilibrium Condition: $p_1 = v_1$.

and

Region-2

Supply side:

$$\begin{aligned} &\min_{x_2} \quad c_2^T x_2 \\ s.t. \quad & A_2 x_2 \geq q_2 \qquad (v_2) \\ & B_2 x_2 \leq b_2 \\ & x_2. q_2 \geq 0 \end{aligned}$$

Demand side: $q_2 = Q_2(p_2)$

Equilibrium Condition: $p_2 = v_2$.

where $\{B_1x_1 \leq b_1 . x_1 \geq 0\}$ and $\{B_2x_2 \leq b_2 . x_2 \geq 0\}$ are the convex polyhedral production feasibility sets described as Z in the EEM.

To illustrate. suppose that the following Figures 2.1 and 2.2 represent the network flows in a model of energy supplies and demands in Region-1 and Region-2. respectively. There may also be non-network constraints in the supply model.

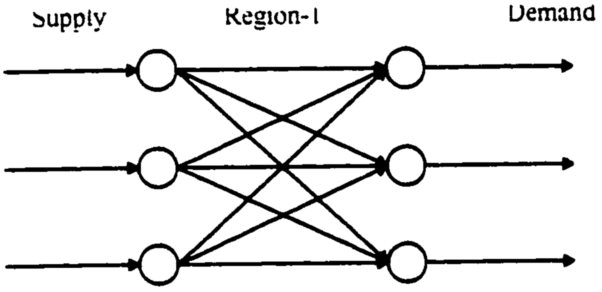

Figure 2.1: The network of energy flows in the submodel of Region-1

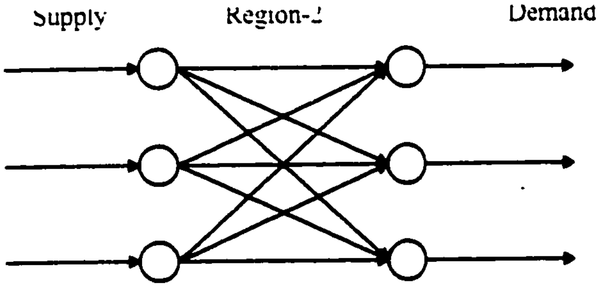

Figure 2.2: The network of energy flows in the submodel of Region-2

If there are some economic activities between these two regional submodels. e.g.. exporting electric power from Region-1 to Region-2 and importing natural gas to Region 1 from Region 2. we will have a new two-region model illustrated in Figure 2.3. where the linking arcs LC11. LC12. LC21. and LC22 are connected with two linking nodes. When these trade links are introduced. the competition between two regions would tend to drive the global supply cost to a minimum. Hence. we get a new two-region model with the following formulation.

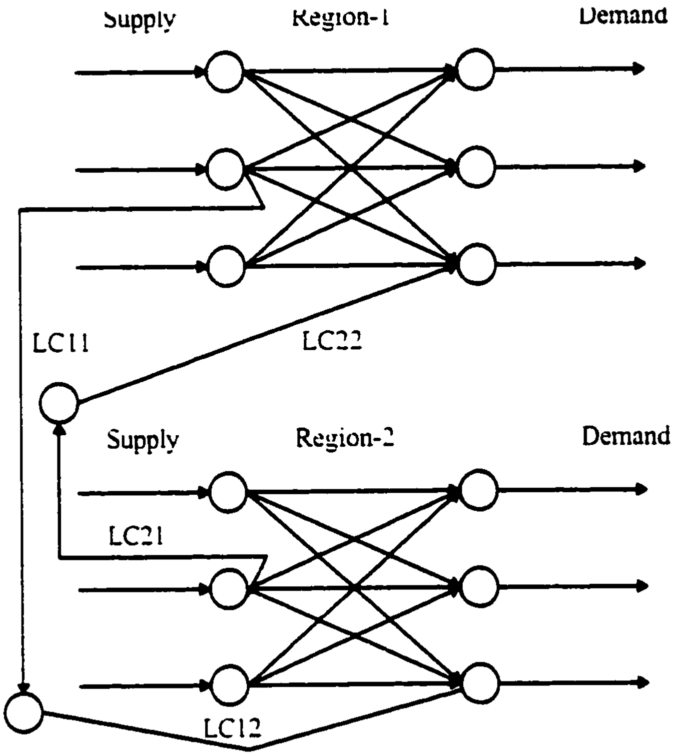

Figure 2.3: The network of the energy flows in the two-region model

Two-region model

Supply side:

$$\min_{x_1, x_2} \quad c_1^T x_1 + c_2^T x_2$$

$$
\begin{aligned}
s.t. \quad & A_1x_1 \geq q_1 \qquad (v_1) \\
& A_2x_2 \geq q_2 \qquad (v_2) \\
& B_1x_1 \leq b_1 \\
& B_2x_2 \leq b_2 \\
& L_1x_1 + L_2x_2 \leq h \\
& x_1, x_2, q_1, q_2 \geq 0
\end{aligned}
$$

Demand side: $q_1 = Q_1(p_1)$. $q_2 = Q_2(p_2)$

Equilibrium Condition: $p_1 = v_1$. $p_2 = v_2$.

where $L_1x_1 + L_2x_2 \leq h$ are the linking constraints. containing the linking arcs and corresponding to the linking nodes.

Apparently. we must consider this two-region model as a whole model. However. if there is an appropriate decomposition method to decompose this two-region model into regional submodels. and therefore the decomposed submodels can be solved individually. then we can use this decomposition method actually to integrate regional submodels. **An appropriate decomposition method can be used as a tool to integrate regional submodels.**

Similarly. when we integrate m regional submodels. we have the following multi-regional economic equilibrium model (MREEM) which will be used to carry out our discussion on the development of the new decomposition methods. For the sake of simplicity and emphasis on the multi-regional structure. indices such as time period t and commodity i. are hidden. leaving only the index r. for regions.

explicitly displayed.

MREEM

Supply side:

$$\min_{x_r,\forall r} \quad c_1^T x_1 + c_2^T x_2 + \ldots + c_m^T x_m$$

$$s.t. \qquad A_r x_r \geq q_r \qquad (v_r) \qquad \forall r \tag{2.10}$$

$$L_1 x_1 + L_2 x_2 + \ldots + L_m x_m \leq h \qquad (\beta) \tag{2.11}$$

$$\left.\begin{array}{llll} B_1 x_1 & & & \leq b_1 \\ & B_2 x_2 & & \leq b_2 \\ & & \ddots \quad \vdots & \vdots \\ & & B_m x_m & \leq b_m \end{array}\right\} \tag{2.12}$$

$$x_r \geq 0. \qquad q_r \geq 0 \qquad \forall r:$$

Demand side: $q_r = Q_r(p_r) \qquad \forall r:$

Equilibrium Condition: $p_r = v_r \qquad \forall r:$

where

c_r = a cost vector for the supply activities in region r:

x_r = a production activity level vector in region r:

v_r = an optimal dual variable vector (shadow price vector) corresponding to the demand requirement constraints $A_r x_r \geq q_r$ in region r:

β = a dual price vector corresponding to the linking economic activity constraints $\sum_{r=1}^{m} L_r x_r \leq h$:

$\sum_{r=1}^{m} L_r x_r \leq h$ represents linking economic activity constraints:

$B_r x_r \leq b_r$ represents supply constraints in region r:

$A_r x_r \geq q_r$ requires that demands are met in region r:

$Q_r(p_r)$ = a vector-valued market demand function defined over prices. p_r in region r.

Notice that the MREEM is an extension of the formulation of the EEM and shares the same properties of the EEM presented in Section (2.1). That is. q_r and v_r are variables. The MREEM is solved when $p_r = v_r$ such that $q_r = Q_r(p_r)$ for all r. Hence. the MREEM is not an optimization model. The regional demand q_r consists of two or more commodity components. The vectors x_r. p_r. c_r. b_r. h. and matrices L_r. B_r have appropriate dimensions. The above constraint set in the supply model is called a **block-angular constraint set**. which is non-separable. Because of the linking constraints. the MREEM cannot break into m separate regional models.

Moreover. according to the definition of equilibrium condition provided in [2]. if the optimal dual variables v_r corresponding to the demand requirement constraint $Ax_r \geq q_r$ in the cost minimizing linear programming supply side is equal to p_r such that $q_r = Q_r(p_r)$. and $Q_r(p_r)$ is strictly monotone for all r. then there exists at most one solution to the MREEM. We shall assume strict monotonicity of demand. in order to ensure uniqueness of the equilibrium.

Variational Inequality Problems

Much recent literature regarding the equilibrium model is based on variational inequality (VI) problems. Based on Sham (1997) [32]. the expression of the MREEM in the formulation of the VI problems is given below.

MREEM-VI

Find $(x^*, q^*) \in K$

$$s.t.\ c^T \bullet (x - x^*) - Q^{-1}(q^*)^T \bullet (q - q^*) \geq 0,$$

$$\forall (x, q) \in K.$$

where K is the constraint set of the supply side of the MREEM and the vectors c, x, q, and Q include all regional components.

The new decomposition methods can be directly used in the MREEM-VI, the variational inequality problems. However, for the sake of simplicity in presentation of how the new decomposition method is derived, the MREEM is used instead of the MREEM-VI.

2.3 Symmetric Equilibrium Model

If the demand function $Q(p)$ satisfies the integrability condition (the matrix of cross-price derivatives of $Q(p)$ is symmetric), then the MREEM is equivalent to the convex optimization model [1] shown below.

$$\min_{x,q} \quad c^T x - \int_0^q Q^{-1}(y)dy$$

$$s.t. \quad (x, q) \in K$$

This kind of model is referred to as a symmetric equilibrium model for the rest of this thesis.

Some researchers claimed that they applied existing LP decomposition principles to their equilibrium models. or the VI problems. However. all their models are symmetric equilibrium models (optimization models). Obviously. if the equilibrium models. or the MREEM. can be converted into a simple optimization problem as above. some existing decomposition methods. e.g.. the Dantzig-Wolfe decomposition method [9]. can be employed directly. Hence. no literature review of decomposition methods for symmetric models (optimization) is presented here.

2.4 Asymmetric Equilibrium Model

In case the demand function $Q(p)$ is non-integrable (asymmetric). the MREEM cannot be converted into an optimization model and this kind of model is referred to as an asymmetric equilibrium model. Since the asymmetric equilibrium model is not an optimization model. we cannot use optimization solution techniques to solve it. The asymmetric equilibrium model can be solved by various approaches. such as fixed-point methods [31]. the PIES method [2]. and nonlinear complementarity programming or variational inequality methods ([20] and [24]). Since the development of the decomposition methods relies upon these solution techniques for subproblems. a brief description of each solution technique is given. The next section reviews existing decomposition methods for asymmetric equilibrium models.

2.4.1 Variational Inequality Methods

Based on Harker and Pang (1990) [13]. we are given a VI problem shown below.

VI

Find $x^* \in K \subseteq R^n$. such that

$$F(x^*) \bullet (x - x^*) \geq 0. \quad \forall x \in K$$

where

F = a given continuous function from K to R^n:

K = a given closed. convex set: and

K is also assumed to be compact and $F(x)$ continuously differentiable.

A general approach for solving the VI problems consists of creating a sequence $x^k \subseteq K$ such that each x^{k+1} solves $VI(K. F^k)$ problem:

$$F^k(x^{k+1}) \bullet (y - x^{k+1}) \geq 0. \forall y \in K.$$

where $F^k(x)$ = an approximation to $F(x)$. which is chosen in a way that makes $VI(K. F^k)$ easy to solve.

Two basic choices for this approximation are that $F^k(x)$ is either a linear or nonlinear function.

For the linear approximations, we have

$$F^k(x) = F(x^k) + A(x^k)(x - x^k)$$

where $A(x^k)$ = an $n \times n$ matrix. Several methods exist which differ in the choice of $A(x^k)$:

$$\begin{aligned} A(x^k) &= \nabla F(x^k) && \text{(Newton's method)} \\ &\approx \nabla F(x^k) && \text{(Quasi-Newton)} \\ &= D(x^k) && \text{(Linearized Jacobi)} \\ &= (L(x^k) or U(x^k)) + \tfrac{D(x^k)}{w^*} && \text{(SOR)} \\ &= \tfrac{1}{2}(\nabla F(x^k) + \nabla F(x^k)^T) && \text{(symmetrized Newton)} \\ &= G && \text{(Projection)} \end{aligned}$$

where

$$\begin{aligned} D(x^k) &\equiv \text{the diagonal part of } \nabla F(x^k). \\ L(x^k) &\equiv \text{the lower triangular part of } \nabla F(x^k). \\ U(x^k) &\equiv \text{the upper triangular part of } \nabla F(x^k). \\ w^* &\equiv \text{a scalar parameter } \in (0.2). \\ G &\equiv \text{a fixed. symmetric. positive definite matrix.} \\ SOR &\equiv \text{Successive Overrelaxation.} \end{aligned}$$

For the nonlinear approximations, the most popular is the nonlinear Jacobi (also called relaxation or diagonalization) algorithm [2]. The basic idea of this algorithm is to extend the linearized Jacobi method by producing a separable nonlinear map at each iteration

$$F^k(x) = (\ldots, F_i^k(x_i), \ldots)^T : R^n \rightarrow R^n$$

where

$$F_i^k(x_i) = F_i(x_1^k, x_2^k, \ldots, x_{i-1}^k, x_i, x_{i+1}^k, \ldots, x_m^k).$$

2.4.2 PIES Method

When the above nonlinear approximation technique is applied to the economic equilibrium model with the formulation of the MREEM. the solution technique for the VI problems actually reduces to the PIES (Project Independence Evaluation System) method. According to Ahn and Hogan (1982) [2]. an illustration of using the PIES method for solving the MREEM is given below. The region index r is hidden for simplicity.

Let

$$q_i(p; p^{k-1}) = Q_i(p_1^{k-1}, p_2^{k-1}, \ldots, p_{i-1}^{k-1}, p_i, p_{i+1}^{k-1}, \ldots, p_I^{k-1}).$$

$$P_i(q_i; p^{k-1}) = Q_i^{-1}(q_i; p^{k-1}).$$

where p_i = price for the i^{th} commodity:

k = iteration index:

i = commodity index:

$P_i(\bullet)$ = inverse of the modified demand function $Q_i(\bullet; p^{k-1})$. i.e.. it gives p_i as a function of q_i. with other commodities' prices fixed.

At the k^{th} iteration of the PIES method. the following optimization problem is solved.

PIES^k

$$\min_{x,q_i} c^T x - \sum_{i=1}^{I} \int_0^{q_i} P_i(y; p^{k-1})dy$$

$$s.t. \ (x, q) \in K$$

where K is the block angular constraint set of the supply side in the MREEM.

The PIES method for the MREEM is:

Step 1: Provide a starting guess of the price vector. p^0.and set $k = 1$:

Step 2: Solve problem PIES^k. constructed by p^{k-1}. to obtain demand estimates q^k: calculate new price estimates p_i^k from $P_i(q_i; p^{k-1})$:

Step 3 : Calculate

$$\theta = \max_i |\frac{p_i^k - p_i^{k-1}}{p_i^{k-1}}|:$$

if $\theta < \delta$. a given tolerance. then go to step 4. otherwise increment the iteration index $k \leftarrow k + 1$. and return to step 2:

Step 4: Terminate with $p^* = p^k$.

2.4.3 Fixed Point Methods

Another computational approach for the asymmetric equilibrium models. which is based on an economic theory. is fixed-point method pioneered by Scarf (1967) [30]. Consider a multi-commodity market in which the consumer's behavior is captured by a demand function $Q(\bullet)$ and the supply sector is described by an inverse supply mapping $P_s(\bullet)$ which is a point-to-set mapping. A typical equilibrium condition of the MREEM is defined as a fixed-point problem:

$$P^* \in P_s(Q(P^*))$$

where $P_s(Q)$ is the image set of the mapping applied to the demand vector Q.

The fixed-point representation readily yields an iterative solution scheme such as

$$P^{k+1} \in P_s(Q(P^k))$$

which is a typical **cobweb** search process. Unfortunately. as can be shown by examining even a one-dimensional example. the cobweb search may diverge in some cases. However. Murphy and Mudrageda (1998) [22] presented a variant of the cobweb method that converges.

Although we also have other convergent fixed point algorithms (see Scarf and Hansen (1973) [31]). their computational efficiency in solving large-scale equilibrium models is doubtful. Ahn (1979) [1] provided a piece of practical evidence showing the inability of the fixed point methods to handle large-scale models.

2.5 Existing Decomposition Methods for Asymmetric Equilibrium Models

Based on the solutions technique. four kinds of decomposition methods have been developed for asymmetric equilibrium models.

2.5.1 Partitionable Decomposition

If the approximation $F^k(x)$ is separable. then each subproblem $VI(K.F^k)$ can be cast into the form of an optimization problem. If the feasible set K of the VI problems is given by

$$\begin{pmatrix} B_1 & 0 & 0 & 0 \\ 0 & B_2 & 0 & 0 \\ 0 & 0 & \ddots & \vdots \\ 0 & 0 & \dots & B_m \end{pmatrix} \bullet \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_m \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{pmatrix}$$

where

$$x_r \geq 0.\ r = 1.\dots.m$$

$$m \geq 1.\ x_r \in R^{n_r}.$$

and the matrices B_r and b_r. $r = 1.\dots.m$. have appropriate dimensions. then the feasible set is the Cartesian product $K = \prod_{r=1}^{m} K_r$. where

$$K_r = \{x_r \in R_{+}^{n_r} |\ B_r x_r = b_r.\ r = 1.\dots.m\}$$

Some researchers. e.g.. Nagurney (1993) [24] shows that if the constraints define a Cartesian product of feasible sets. the VI problems with a linear or nonlinear approximation function $F_i^k(x)$ can be decomposed into m subproblems. The most common linear approximation function to be used for decomposition purpose is SOR. For a comprehensive review. the reader may refer to [28].

The separable structure of the feasible set K is the main assumption of the Partitionable Decomposition method. Obviously. this method is not of interest to us because the multi-regional model has a feasible set that is block-angular. not a Cartesian product.

2.5.2 Transfer Decomposition

Hearn (1984) [14] has developed the Transfer Decomposition scheme for symmetric traffic assignment problems. This scheme is to partition a network so that the original problem is transformed into two problems: a master problem and a subproblem. Barton. Hearn and Lawphongpanich (1989) [4] show that this technique is equivalent to a generalized Benders decomposition of the original equilibrium problem. Lawphongpanich and Hearn (1990) [17] extended the above technique to asymmetric models. However. this extension requires a point-to-set mapping imposed in the Master problem. which is very difficult to solve [13] due to the nonconvexity of the transformed feasible region.

The bigger the number of the subproblems. the harder the master problem is to solve. because the number of the point-to-set mappings imposed in the master problem is equal to the number of the subproblems in the model. Thus. this scheme not only does not provide an efficient decomposition method. but also destroys the mathematical formulation of the original equilibrium model. Moreover. this method is not similar to the Dantzig-Wolfe decomposition principle in which the original problem is not transformed to another type of problem. That is. if the original problem can be solved by one algorithm (e.g. the simplex algorithm for LPs). then the decomposed problems (including Master and sub-problems) can be solved by the same algorithm.

2.5.3 Simplicial Decomposition

This method is usually applied to the traffic assignment problem. Lawphongpanich and Hearn (1984) [16]. and Pang and Yu (1984) [26] show that it can be an effective computational technique for large-scale problems. It also can be used in both symmetric and asymmetric models.

The main idea of the Simplicial Decomposition method is derived from column generation in linear programming. It is a scheme for algorithmically generating profitable variables in a problem. The algorithmic principle consists of two main steps. In the first. the original problem is solved over the set of known variables (restricted master problem). In the second. the solution to this master problem is the basis for the formulation of a subproblem. which is solved to generate variables that may improve the restricted master problem solution. However. because the subproblem is still subject to the whole feasible set. i.e.. no decomposition in the block-angular feasible sets. this scheme is not suitable for the multi-regional models either.

2.5.4 Cobweb-Decomposition

In the absence of appropriate demand-supply decomposition methods. some researchers (see Murphy et al. (1988) [23]. Murphy and Mudrageda (1998) [22]. Wagner (1980) [34]. Mansur and Whalley (1982) [19]. and Bueler (1997) [7]) adopted the cobweb algorithm for their tailor-made demand-supply decomposition methods. By the cobweb algorithm. the supply model (or demand model) estimates a price vector. given the quantity offered by the demand model (or supply model). and passes the price information to the demand model (or supply model). This process continues in an iterative manner until the price and quantity approximations stop changing. The general principle of their decomposition methods is therefore based on decomposing an economic equilibrium model into a supply model and a demand model. but the same technique could be used for regional decomposition. By means of the cobweb algorithm. price and quantity pairs are passed between models as a way to decompose the model. Unfortunately. as can be shown by examining even a one-dimensional example. the cobweb algorithm may diverge in some cases. We refer to this type of decomposition method as the cobweb-decomposition method.

2.6 Summary

Because of violating the integrability condition. some equilibrium models cannot be converted into optimization models. These equilibrium models are called asymmetric equilibrium models in this dissertation. The asymmetric equilibrium models can be solved by variational inequality methods. the PIES method and Fixed point methods. Four decomposition methods have been developed for computational efficiency.

However. Partitionable Decomposition relies upon the separability of the feasible set to carry out the decomposition in each iteration of the solution method for the VI problems. Transfer Decomposition makes the decomposed model very difficult to solve due to the nonconvexity of the transformed feasible region. In Simplicial Decomposition. the subproblems are still subject to the whole constraints set. i.e.. there is no decomposition in the block-angular feasible sets. Cobweb-Decomposition may diverge. Hence. these methods are not suitable for our integration purpose in which regional models are to be integrated as a whole multi-regional economic equilibrium model.

Chapter 3

A New Decomposition Method

Since at each iteration of the VI methods or the PIES method. an approximated solution can be obtained by solving an optimization problem. we can apply an existing decomposition principle to decompose such an optimization problem. We refer to this type of decomposition scheme as a ***sequential optimization decomposition scheme***. in which the decomposition principle is employed within each sequential iteration. Obviously. this type of decomposition approach is not of interest in this dissertation because the integration of the regional models depends on the type of solution procedure for equilibrium models. We are interested in decomposition schemes which depend only on the structure of the links among regions.

A new structural decomposition scheme is developed. in which any equilibrium seeking algorithm is used to solve equilibrium subproblems individually - we use the PIES method here. The original MREEM is to be decomposed into regional economic equilibrium models structurally. just as existing Linear Programming

(LP) decomposition principles decompose a large-scale LP into several smaller LPs. i.e. the same class of model as the original. In our example. all these regional EEMs are to be solved by the PIES method individually within each decomposition iteration.

3.1 The new decomposition method, for MREEM

The new decomposition method for the MREEM relies on convex combinations and column generation techniques as in the decomposition principle of linear programming. There are three main steps in the new decomposition method.

Step 1: Decompose the LP supply side of the MREEM (page 14) into subproblems (or submodels - the term "subproblems" is used here instead of "submodels" due to the traditional terminology in discussions of the decomposition principles) by an existing LP decomposition principle. Each subproblem includes the corresponding regional objective function. regional demand requirement constraints (2.10) and regional supply constraints from (2.12). Depending on the decomposition principle. there may also be a master problem.

Step 2: Attach the regional demand side to the corresponding subproblems and the master problem such that all subproblems and the master problem are transformed into equilibrium subproblems and equilibrium master problem.

Step 3: Solve the decomposed equilibrium model by exchanging the price information for the linking constraints and quantity information among all equilibrium subproblems and the master problem in an iterative manner until an equilibrium point is found. The pattern of the decomposition-based information exchange follows that of the adopted decomposition principle. At each iteration. the master problem and all equilibrium subproblems can be solved by the PIES method (or other methods). Note that different decomposition principles can be adopted by the new decomposition method. which leads to different algorithms.

3.2 An illustration, with the Dantzig-Wolfe principle

To illustrate the new decomposition method. the Dantzig-Wolfe [9] decomposition principle is adopted. For simplicity. the following two-region model is used. and we have one equilibrium master problem and two regional equilibrium subproblems.

Two-region model

Supply side:

$$\min_{x_1, x_2} \quad c_1^T x_1 + c_2^T x_2$$

$$s.t. \qquad A_1 x_1 \geq q_1 \qquad (v_1)$$

$$A_2 x_2 \geq q_2 \qquad (v_2)$$

$$B_1 x_1 \leq b_1$$

$$B_2 x_2 \leq b_2$$

$$L_1 x_1 + L_2 x_2 \leq h \qquad (\beta)$$

$$x_1, x_2, q_1, q_2 \geq 0$$

Demand side: $q_1 = Q_1(p_1)$, $q_2 = Q_2(p_2)$

Equilibrium Condition: $p_1 = v_1$, $p_2 = v_2$.

3.2.1 Illustration of Step 1

Applying the Dantzig-Wolfe decomposition principle to the LP supply side of the Two-region model, i.e. for fixed q_1 and q_2, we have one restricted master problem and two sub-problems in the supply side at the k^{th} decomposition iteration.

Supply side at the k^{th} iteration:

Restricted master problem

$$\min_{\lambda_1^k, \lambda_2^k} \quad c_1^T X_1^k \lambda_1^k + c_2^T X_2^k \lambda_2^k$$

$$s.t. \quad A_1 X_1^k \lambda_1^k \geq q_1 \qquad (v_1)$$

$$A_2 X_2^k \lambda_2^k \geq q_2 \qquad (v_2)$$

$$L_1 X_1^k \lambda_1^k + L_2 X_2^k \lambda_2^k \leq h \qquad (\beta)$$

$$e^k \lambda_1^k = 1; \quad e^k \lambda_2^k = 1$$

$$\lambda_1^k, \lambda_2^k, q_1, q_2 \geq 0$$

where

$$X_r^k = (x_r^1, x_r^2, \ldots, x_r^k), \qquad r = 1, 2;$$

x_r^f = supply subproblem solution from iteration f. $r = 1.2$:

$\lambda_r^k = (\lambda_{r1}.\lambda_{r2}.\dots.\lambda_{rk})^T. r = 1.2$:

$e^k = (1.1.\dots.1)$ the unit k-vector.

Subproblems

$$\min_{x_1} \quad (c_1 - \beta L_1)^T x_1$$

$$s.t. \quad A_1 x_1 \geq q_1: \quad B_1 x_1 \leq b_1: \quad x_1 \geq 0:$$

and

$$\min_{x_2} \quad (c_2 - \beta L_2)^T x_2$$

$$s.t. \quad A_2 x_2 \geq q_2: \quad B_2 x_2 \leq b_2: \quad x_2 \geq 0.$$

3.2.2 Illustration of Step 2

If we were to extend the above decomposed supply side to an equilibrium model. we would need to include the following conditions.

Demand side: $q_1 = Q_1(p_1)$. $q_2 = Q_2(p_2)$

Equilibrium Condition: $p_1 = v_1$. $p_2 = v_2$.

If we distribute the demand functions and equilibrium conditions appropriately among subproblems and the restricted master problem. we define the following equilibrium problems.

Restricted equilibrium master problem at the k^{th} iteration (**REMP^k**):

Supply side:

$$\min_{\lambda_1^k, \lambda_2^k} \quad c_1^T X_1^k \lambda_1^k + c_2^T X_2^k \lambda_2^k$$

$$s.t. \quad A_1 X_1^k \lambda_1^k \geq q_1 \qquad (v_1)$$

$$A_2 X_2^k \lambda_2^k \geq q_2 \qquad (v_2)$$

$$L_1 X_1^k \lambda_1^k + L_2 X_2^k \lambda_2^k \leq h \qquad (\beta)$$

$$e^k \lambda_1^k = 1; \qquad e^k \lambda_2^k = 1$$

$$\lambda_1^k, \lambda_2^k, q_1, q_2 \geq 0$$

Demand side: $q_1 = Q_1(p_1)$, $q_2 = Q_2(p_2)$

Equilibrium Condition: $p_1 = v_1$, $p_2 = v_2$.

Equilibrium sub-problem 1 (**ESP-1**)

Supply side:

$$\min_{x_1} \quad (c_1 - \beta L_1)^T x_1$$

$$s.t. \quad A_1 x_1 \geq q_1 \qquad (v_1)$$

$$B_1 x_1 \leq b_1; \qquad x_1 \geq 0; \qquad q_1 \geq 0$$

Demand side: $q_1 = Q_1(p_1)$

Equilibrium Condition: $p_1 = v_1$.

Equilibrium sub-problem 2 (**ESP-2**):

Supply side:

$$\min_{x_2} \quad (c_2 - \beta L_2)^T x_2$$

$$s.t. \quad A_2 x_2 \geq q_2 \qquad (v_2)$$

$$B_2 x_2 \leq b_2; \; . \; x_2 \geq 0; \quad q_2 \geq 0$$

Demand side: $q_2 = Q_2(p_2)$

Equilibrium Condition: $p_2 = v_2$.

Since any proposals generated by equilibrium subproblems satisfy the constraints $B_r x_r \leq b_r$. we can leave these constraints out of the equilibrium master problem. Consequently. we have one restricted equilibrium master problem REMP^k and two equilibrium subproblems ESP-r. The equilibrium master problem has few constraints. compared to the whole model because the dimensions of q_r and h are small for the models that we consider.

Decomposition may result in unboundedness in the subproblems. If this is the case. upper bounds can be imposed on all variables to prevent the subproblems from being unbounded. This technique is used in our test models.

3.2.3 Illustration of Step 3

According to the adopted decomposition principle. the Dantzig-Wolfe decomposition principle. we have the following decomposition-based information exchange. All equilibrium subproblems are solved by the PIES method (or another algorithm). with a given equilibrium price vector β to provide a new equilibrium proposal. The restricted master problem. solved by the PIES method (or another algorithm). estimates a new price vector β with all the accumulated proposals. This iterative process will be terminated by a stopping condition - see the next section.

It is easy to extend the above development of the new decomposition method to the MREEM. We assume that we have the decomposed MREEM and carry out the following discussions. Figure 3.1 illustrates the decomposed MREEM. which consists of one equilibrium master problem and m equilibrium subproblems.

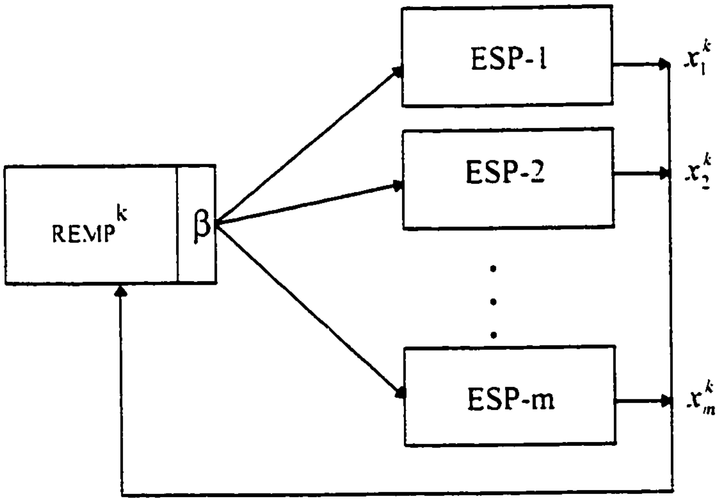

Figure 3.1: The decomposed MREEM by Dantzig-Wolfe principle

3.3 Stopping and convergence conditions

As discussed in Section 2.2. if we assume that $Q_r(p_r)$ is monotone for all r. there exists at most one solution to the MREEM. Based on this uniqueness assumption. we have the following stopping conditions.

3.3.1 Stopping conditions

After RMEP^k has been solved. we have an equilibrium for a model whose supply side is more restricted than in the full model MREEM. i.e.. the supply cost may be greater than it would be for the unrestricted supply of MREEM. If it happens that the supply cost from RMEP^k actually is the minimum that could be achieved in the unrestricted supply of MREEM. then RMEP^k has found an equilibrium for the full model MREEM. This reasoning leads to a general stopping condition. which we label "sufficient" because. if it is satisfied. then we may stop the algorithm because we have found the solution.

> **Sufficient stopping condition**
>
> Given $\hat{q}_r$ and $\hat{x}_r = X_r^k \lambda_r^k$ from RMEP^k. if the cost of the LP supply side of the MREEM is minimized by $\hat{x}_r$. then $\hat{q}_r$. $\hat{x}_r$ is the unique solution to MREEM.

However. directly solving the LP supply side of the MREEM is not consistent with our aim to decompose by region. The stopping condition for the Dantzig-Wolfe decomposition principle of LPs [9] can be used to overcome this difficulty. We can investigate all equilibrium subproblems. rather than the equilibrium master problem. to check that there are no more proposals that can reduce the cost of the equilibrium master problem. That is. we can use equilibrium subproblems to check if the sufficient stopping condition is satisfied and therefore the MREEM is solved.

Because direct implementation of the sufficient stopping condition would require an LP calculation for each region at each iteration. we use a simpler calculation first which allows us to avoid the LP calculations for most iterations. The simple calculation is the basis of a necessary stopping condition: if the equilibrium has been found. then the condition must be satisfied. or conversely (which is how we use the condition). if the condition is violated. then the equilibrium has not been found and the iterations must continue.

The necessary stopping condition of the equilibrium solution is based on the solution technique of the PIES method. in which the equilibrium point is found by solving a sequence of nonlinear programs as discussed in Section 2.4.2. If RMEP^{k-1} finds the equilibrium of the MREEM. then further subproblem proposals cannot reduce the supply costs. and. by uniqueness of the equilibrium. the same equilibrium must be found by RMEP^{k}. The last nonlinear programs in the PIES sequences for RMEP^{k-1} and RMEP^{k} must have the same objective functions and the same objective values. We summarize the condition below.

Necessary stopping condition

If the equilibrium has been found by RMEP^{k-1}. then the difference between the k^{th} and $(k-1)^{th}$ objective values of the master problem is zero. Conversely. if the difference is not zero. then the equilibrium has not been found. and the iterations must continue.

However. the condition of the same objective values on successive iterations is not sufficient to terminate the calculation because the master problem is an equilibrium model: two successive solutions to the restricted master problem may both fail to be the equilibrium of MREEM. yet it is possible that their final PIES objective values could. by chance. be equal. Nevertheless. we have found. in all tests so far. that stopping when the difference in successive objectives is sufficiently small does yield the unique solution of the MREEM.

3.3.2 Convergence consideration

The convergence properties are not as easy to investigate as for the Dantzig-Wolfe decomposition algorithm for LPs. The convergence theorem of the Dantzig-Wolfe algorithm is based on the finite number of extreme points generated by subproblems. However. in the new decomposition method. the equilibrium subproblems will generate extreme points of the x_r feasible sets. but these sets change because q_r changes in the iterations. Hence. other approaches to convergence theorems are needed. We have not yet found a theorem that guarantees convergence. so we must leave this for future research. As we report in Section 3.7. the algorithm does converge in all tests. so far.

3.4 The algorithm, with the Dantzig-Wolfe principle

When the new decomposition method adopts the Dantzig-Wolfe decomposition principle. we have the following algorithm for solving the decomposed MREEM.

Step 1: Set $\beta = \beta^0$ a guess provided by modelers: solve all ESP-r: index the proposal with $k = 1$.

Step 2: Solve the REMP^k. If $k \geq 2$. go to Step 3: else go to Step 4.

Step 3: Check whether to continue. using the necessary stopping condition. If yes. go to Step 4: else go to Step 5.

Step 4: Solve all ESP-r with equilibrium dual price β from Step 2: increment $k = k + 1$: index the proposal with k: go to Step 2.

Step 5: Check if the MREEM is solved with the sufficient stopping condition. i.e. solve m LPs defined as ESP-r with fixed demand q_r and β provided from Step 2. and apply the Dantzig-Wolfe stopping condition. If yes. stop. Else. go to Step 4.

As mentioned in Section 3.3.1. the sufficient stopping condition requires the solution of an LP for each equilibrium subproblem. and the necessary stopping condition only compares two numbers. Obviously. checking the sufficient stopping condition in each decomposition iteration is inefficient. Therefore. the necessary stopping condition is checked first and then the sufficient stopping condition is checked only when warranted. The algorithm is presented as a flowchart in Figure 3.2. with the information exchange indicated by dashed lines.

The new decomposition method with Dantzig-Wolfe decomposition principle can be interpreted as a mathematical representation of decentralized planning. The coordinating research group (the equilibrium master problem) posts commodity prices for the linking (e.g.. trading) resources available to regional research groups (the equilibrium subproblems). and regional research groups submit equilibrium plans (proposals) based on these prices. That is. the coordinating research group makes a compromise equilibrium plan based on all proposals submitted so far. and posts new prices to get more information from regional research groups. The

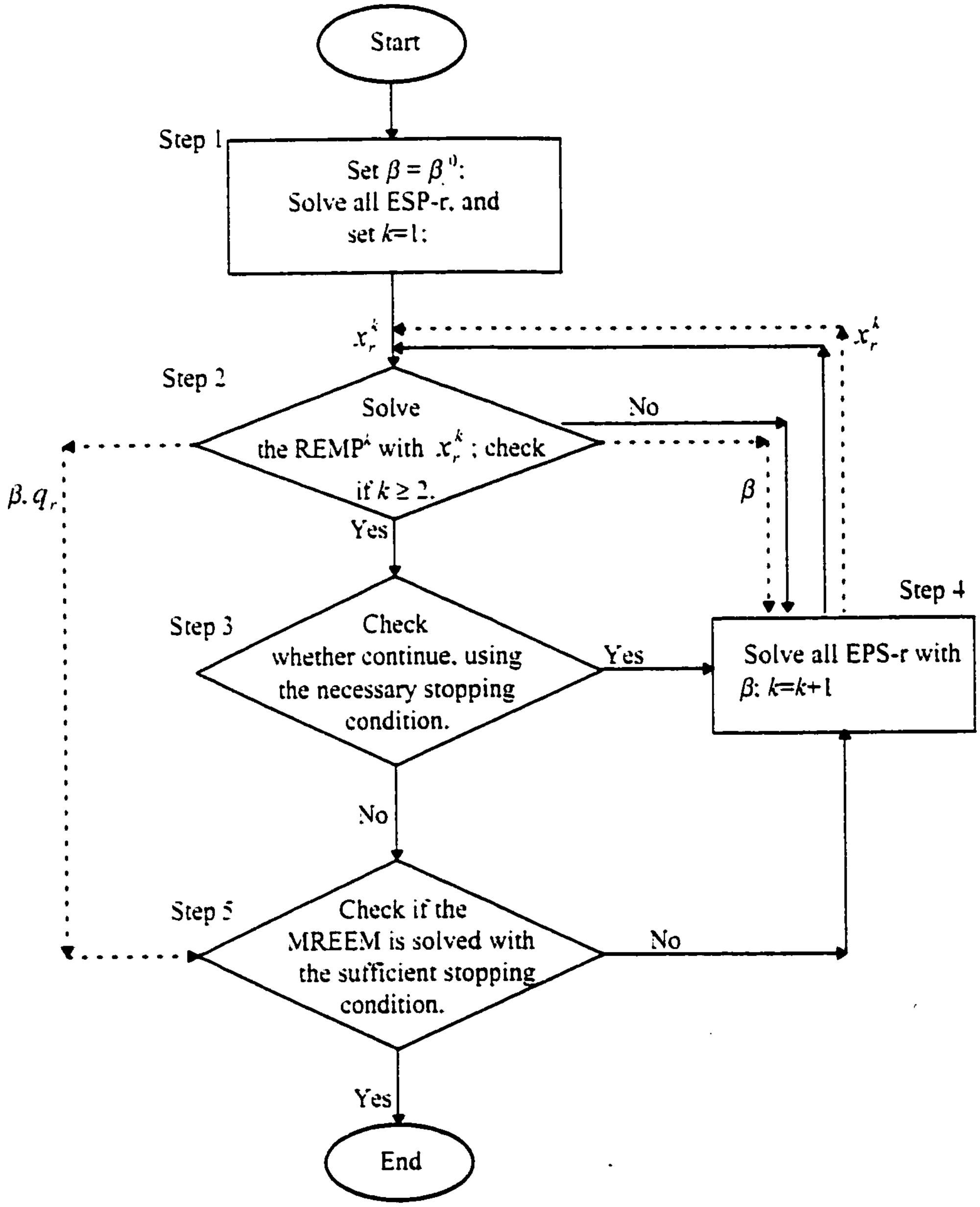

Figure 3.2: The flowchart of the algorithm. with Dantzig-Wolfe principle

coordinating research group is supposed to have complete information about the dependence of demands on prices. but its incomplete information about supply conditions gradually improves as the iterations proceed.

3.5 The new decomposition method in VI

Since the MREEM can be expressed in a VI form. we have

MREEM-VI

Find $(x^*. q^*) \in K$

$$s.t. \quad c^T \bullet (x - x^*) - Q^{-1}(q^*)^T \bullet (q - q^*) \geq 0. \forall (x. q) \in K.$$

where

where K is the constraint set of the supply side of the MREEM and the vectors $c. x. q.$ and Q include all regional components.

By using the new decomposition method with the Dantzig-Wolfe decomposition principle. the MREEM-VI can be decomposed into one equilibrium master problem Master-VI. and equilibrium subproblems SP$_r$-VI for $r = 1. \ldots . m$.

Master-VI

Find $\lambda_r^{*k}. q_r^* \in S_m^k$

$$s.t. \quad \sum_{r=1}^{m} [c_r^T X_r^k (\lambda_r^k - \lambda_r^{*k}) - Q_r^{-1}(q_r^*)^T (q_r - q_r^*)] \geq 0. \quad \forall (\lambda_r^k. q_r) \in S_m^k$$

where

$$S_m^k = (\lambda_r, q_r)$$

such that

$$A_r X_r^k \lambda_r^k \geq q_r, \qquad \forall r$$

$$\sum_{r=1}^{m} L_r X_r^k \lambda_r^k \leq h, \qquad (\beta)$$

$$e^k \lambda_r^k = 1, \qquad \forall r$$

$$\lambda_r^k, q_r \geq 0 \qquad \forall r$$

SP$_r$-VI

Find $x_r^*, q_r^* \in S_r$

$$s.t. \quad (c_r - \beta L_r)^T (x_r - x_r^*) - Q_r^{-1}(q_r^*)^T (q_r - q_r^*) \geq 0, \qquad \forall (x_r, q_r) \in S_r$$

where

$S_r = \{(x_r, q_r) | A_r x_r \geq q_r, \quad B_r X_r \leq b_r, \quad x_r, q_r \geq 0\}$; and

β = dual variables which can be obtained from solutions of the Master-VI.

The above formulae show that the equilibrium master problem and subproblems can be expressed as the VI problem. These separated VI problems can be solved by any appropriate solution techniques.

The new decomposition method can be extended to the VI problems in a more general way. It can be generalized for VI problems consisting of ***easy*** variables (x_r in this thesis) and ***hard*** variables q_r. The distinction between ***easy*** and ***hard*** variables is that only the hard variables make the VI problems into asymmetric equilibrium problems. If the constraint set links the x_r vectors for different values of r. but not the q_r vectors. then the new decomposition method provides a way to decompose a VI problem.

3.6 Other decomposition principles

One of the characteristics of the new decomposition method is the flexibility to adopt other appropriate decomposition principles. In this section. another decomposition principle. the Lan-Fuller decomposition principle (see Lan and Fuller [15]) is adopted. as an illustration.

For simplicity. the two-region model is used again. The Lan-Fuller decomposition principle divides the LP supply side of the two-region model into two subproblems instead of one master and two subproblems by the Dantzig-Wolfe principle. Hence. following the three main steps of the new decomposition method. the Two-region model can be decomposed into only two equilibrium subproblems. i.e.. no equilibrium master problem. Each subproblem accumulates either primal or dual proposals from the other subproblem. building a compact approximation of the other subproblem.

The following decomposition structure is obtained at the k^{th} decomposition iteration.

SUB1^k

Supply side:

$$\min_{x_1,\theta} c_1 x_1 - \theta$$

$$s.t. \quad A_1x_1 \geq q_1 \qquad (v_1)$$

$$B_1x_1 \leq b_1$$

$$\pi_3^iL_1x_1 + \theta \leq \pi_2^iq_2 + \pi_3^ib_2 + \pi_4^ih \qquad \forall i \qquad (\mu^i)$$

$$x_1.q_1.q_2 \geq 0. \quad \theta \text{ is unrestricted.}$$

Demand side: $q_1 = Q_1(p_1)$: $q_2 = Q_2(p_2)$

Equilibrium condition: $p_1 = v_1$: $p_2 = \sum_{i=1}^{k} \mu^i\pi_2^i$.

SUB2^k

Supply side:

$$\min_{\lambda_1^k.x_2} c_1X_1^k\lambda_1^k + c_2x_2$$

$$s.t. \quad A_1X_1^k\lambda_1^k \geq q_1 \qquad (\pi_1)$$

$$A_2x_2 \geq q_2 \qquad (\pi_2)$$

$$B_2x_2 \leq b_2 \qquad (\pi_3)$$

$$L_1X_1^k\lambda_1^k + L_2x_2 \leq h \qquad (\pi_4)$$

$$e^k\lambda_1^k = 1. \quad \lambda_1^k.x_2.q_1.q_2 \geq 0$$

Demand side: $q_1 = Q_1(p_1)$: $q_2 = Q_2(p_2)$

Equilibrium condition: $p_1 = \pi_1$: $p_2 = \pi_2$.

Figure 3.3 shows the decomposition information exchange between the two equilibrium subproblems. At the k^{th} iteration. the proposal x_1^k is passed to SUB2 from SUB1 as the dual prices (π_2. π_3. and π_4) are passed to SUB1 from SUB2. More-

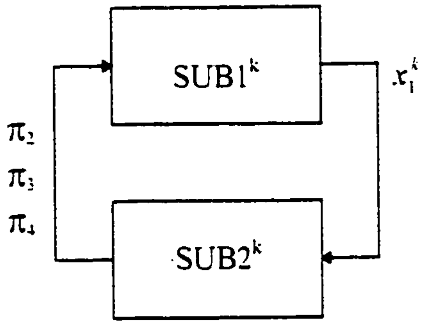

Figure 3.3: The decomposed Two-region model by Lan-Fuller principle

over. the Lan-Fuller decomposition principle can be extended to MREEM. using procedures defined by Park [27].

3.7 Empirical results

In order to report the computational behavior of the new decomposition method. three asymmetric equilibrium models are used for tests.

One of the test models. adopted from [33]. named Small. is a small. two-region economic equilibrium model. There are 6 q-variables. 28 x-variables (including 4 linking variables) and 14 constraints (including 2 linking constraints).

Figure 2.3 shows the structure of the Small model. The supply model is a network flow model. so the constraints are flow balance constraints. corresponding to nodes in Figure 2.3.

Four linking variables (LC11. LC12. LC21 and LC22) with upper bounds are added to what is otherwise two copies of the EEM of [33]. In the Small model. two linking nodes for four linking variables are introduced so that we can express the mathematical linking constraints and separate the regional variables and constraints more easily. The demand functions of the Small model are in the form of

$$q_{rj} = a_{rj} + \sum_{i=1}^{3} b_{rji} \times p_{ri}$$

where

r = index of region = 1. 2:

j = index of commodity = 1. 2. 3:

i = alias index of j:

q_{rj} = demand of commodity j in region r:

p_{ri} = price of commodity i in region r: and

a_{rj} and b_{rji} are constant parameters for all $r. j. i.$

The second one. named CEM. is a realistic two-region energy equilibrium model. the Canadian Energy Model. retrieved and modified from Wu and Chung (1997) [35]. The Canadian Energy Model consists of 14 q-variables. 149 x-variables (including 4 linking variables). and 120 constraints (including 2 linking constraints). per period. Each of the two periods corresponds to a three-year duration. for a total time span of six years from 1986 to 1992.

The third one. named CAN-US. is a realistic 6-region energy equilibrium model. the Canadian-USA Energy Model. retrieved and modified from Chung et al. (1997) [8]. The Canadian-USA Energy Model consists of 43 q-variables. 529 x-variables (including 92 linking variables). and 406 constraints (including 46 linking constraints).

per period. Each of the two periods corresponds to a three-year duration. for a total time span of six years from 1986 to 1992.

For the Small model. the new decomposition method was coded with the Lan-Fuller principle and the Dantzig-Wolfe principle. The programming codes of the Small model are included in Appendix. The Canadian Energy Model and the Canadian-USA Energy Model were coded with the Dantzig-Wolfe principle only. All models were coded into GAMS [6] programs and were executed in an IBM RS/6000 workstation. We use GAMS for the coding language because it can provide access to the nonlinear programming solver MINOS 5.3 from a procedural language. and because the coding effort is much less than for a program in FORTRAN or C. However. there is a major drawback of using GAMS: each call to MINOS. to solve an optimization problem in a PIES sequence. must be preceded by GAMS generating the entire model for input to MINOS. even though successive models are very similar and every optimization is started from the last solution. This repeated model generation. which adds greatly to the total solution time. could be avoided in a carefully written implementation of the decomposition in FORTRAN or C.

The reference method is the PIES method. used to solve the original model with no decomposition. The reference method was coded for the test models so that we can have reference results for evaluating the accuracy and speed of the new decomposition method. All equilibrium subproblems and the master problems of the test models and all reference models are solved by the PIES method with the same convergence tolerance. In the decomposition method. artificial variables with large cost coefficients are added in each linking constraint. and upper bounds are imposed on all variables which do not already have any upper bounds: these measures are taken to prevent infeasibility or unboundedness caused by decomposition. The first guess of the linking prices, β^0, was the zero vector, for all tests.

To help evaluate the new decomposition method, we construct measures of accuracy and effort, namely "D-ITER", "TIMEUSED", "MAX%DIFF", and "ELAPSED", defined as follows.

D-ITER = the number of decomposition iterations used by the new method;

TIMEUSED = the total solver time used to solve the test model, as provided by the MINOS solver (i.e., this excludes GAMS model generation time), measured in seconds;

MAX%DIFF = the maximum percent difference, over all prices and demands between solutions from the new method and the reference method. The following formulation was used.

$$MAX\%DIFF = 100 * max\{max_{ir}(\frac{|p'_{ir} - p_{ir}|}{p'_{ir}}), max_{ir}(\frac{|q'_{ir} - q_{ir}|}{q'_{ir}})\}.$$

where p'_{ir} and q'_{ir} (p_{ir} and q_{ir}) are the solution of prices and demands respectively found by the reference method (the new method).

ELAPSED = the elapsed times of solving the test models under GAMS (i.e. including GAMS model generation time), measured in seconds.

Table 3.1. presents the results from the test models and the reference models.

Model	Method	D-ITER	TIMEUSED	MAX%DIFF	ELAPSED
Small	LF	4	0.39	0	39
	DW	12	1.25	0	69
	REF	—	0.04	—	3
CEM	DW	19	10.55	0.233	417
	REF	—	1	—	58
CAN-US	DW	220	14820.79	0.35	> one day
	REF	—	10.8	—	256

Table 3.1: The computational results of the new decomposition method.

The maximum difference between the new decomposition method and the reference method is acceptable. and the solver time used by the new method is acceptable (except perhaps for the largest model) although longer than the reference method. The very long elapsed time. especially for the largest model. shows the importance of avoiding repeated model generation if these were to be coded for practical use.

Results concerning the accuracy of the new decomposition method are very encouraging. Although the solution time of the new decomposition method is longer than for the reference method. the solution time is acceptably short. considering expected reductions in the modeling time. Furthermore. as Murphy and Mudrageda (1998) also discussed. the solution time would not be serious because the models (i.e.. equilibrium master problem and subproblems in here) are typically distributed over workstations that could be run in parallel.

One may notice that CAN-US took many more iterations than CEM. The reason is that the network structures of the oil and gas sectors in CAN-US are not in a natural regional form. Consequently. there are many linking constraints and variables: intuition suggests that many decomposition steps are required to solve CAN-US. Dirickx and Jennergren (1979) [10] claimed. based on tests. that the number of decomposition iterations increases with the number of linking constraints. There are 2 linking constraints out of 120 constraints. per period. for CEM. However. CAN-US consists of 46 linking constraints out of 406 constraints. per period. The density of the linking constraints of CAN-US is much higher than that of CEM.

Since the discussion of Dirickx and Jennergren (1979) [10] is for LPs. we conducted a test on the Small model to verify the effect of the number of links on the number of iterations. When we added one more linking constraint to Small-DW. it took 16 iterations (4 more than the original model). This test. together with the results for CEM and CAN-US. illustrate that if the MREEM is modeled in a natural regional way with few linking variables and constraints. the solution time is acceptable compared with the modeling time. Moreover. even if the MREEM is not in a natural regional form like CAN-US. the new decomposition method can still solve it with perhaps a long solution time.

The Lan-Fuller method takes fewer iterations and less time than the Dantzig-Wolfe method. on the small test problem. The discussion of selection of β^0 can be found in Chapter 5. Generally. a good first choice of β^0 can reduce the number of decomposition steps.

3.8 Summary

In this Chapter. a new decomposition method has been developed for non-optimization multi-regional economic equilibrium models. Modelers can use the new decomposition method to integrate several well-developed regional economic equilibrium models without the asymmetric restriction from the demand side. The empirical tests have indicated great promise to use the new decomposition method for the purpose of integration. in which regional economic equilibrium models can be modeled individually and then integrated as a multi-regional model. The new method can be viewed as a general procedure for decomposing asymmetric equilibrium models (asymmetric variation inequality problems) which consist of a special structure. e.g.. block-angular linear constraint set. with a few *hard* variables (which cause the asymmetry) and many *easy* variables.

Different decomposition principles can be adopted by the new decomposition method. which leads to different versions of the algorithm. Future research will focus on the computational behavior and convergence characteristics of the new decomposition method for different decomposition principles. On the other hand. efficient implementations of the new decomposition method. e.g.. parallel implementation. is another important issue both for improving the computational speed and allowing integration of regional models which are on different computers connected by a network.

Chapter 4

A New Decomposition Method for MREEM with Geometric Distributed Lag Demand

A class of multi-regional economic equilibrium models. energy equilibrium models. was developed and used for energy policy analyses in the 1970s due to the oil crisis. Later in the 1980s. because of the inter-relationship between the use of energy and the problems with emissions of air pollutants. environmental features were added to the models. Since influence of environmental pollution and the consequences of the environmental protection policies accumulate with time. it is necessary that models be multi-period. In many such multi-period models. demands are functions only of current period prices. However. consumers' adjustments to price changes do not occur instantaneously. so such models can give quite incorrect estimates of pollutants. When prices change in a period. demands often change most significantly in later periods. Such time-lagged effect has drawn some researchers' attention to modify their models. See Wu and Fuller (1995) [36] for further discussions.

Wu and Fuller (1996) [37] introduced a Geometric Distributed Lag (GDL) structure to the demand side of an energy equilibrium model for simulating the time-lagged effect. Because of convergence difficulties of finding the equilibrium of such energy-equilibrium models. the Decoupling algorithm was developed. As explained in detail below. the inter-period price elasticities are eliminated (thus the term "Decoupling") and the current period price elasticities are adjusted in a compensating way at each iteration. For simplicity. the MREEM with GDL demand side is called MREEM-GDL model for the rest of this dissertation.

This chapter is motivated by the MREEM-GDL model and the Decoupling algorithm. The application of the new decomposition method in Chapter 3 to the MREEM-GDL is examined. Tests show that another new decomposition method of is required because the new decomposition method of Chapter 3 fails to converge in solving the MREEM-GDL.

4.1 The model and solution techniques

The difference between the MREEM and the MREEM-GDL is in the demand side. Without considering the time-lag effect. Fuller and Luthra (1990) [11] used the following demand functions to represent the demand side in the MREEM.

Non-GDL Demand

$$q_{itr} = a_{itr} p_{itr}^{-b_{iir}} \prod_{j \in H_{ir}} p_{jtr}^{-b_{ijr}} \tag{4.1}$$

where

q_{itr} = the demand of commodity i in period t. region r:

a_{itr} = the constant factor in demand function of commodity i in period t. region r:

H_{ir} = the set of indices of other demand variables in the same group as q_{itr}:

p_{itr} = the price of commodity i in period t. region r:

b_{iir} = the own-price elasticity of demand for commodity i. region r:

b_{ijr} = the cross-price elasticity of demand for q_{itr} with respect to the price p_{jtr} $(j \in H_{ir})$.

Notice that the own and cross-price elasticities $(b_{iir} .b_{ijr})$ are independent of time and therefore the demands are functions only of current period prices.

On the other hand. Wu and Fuller (1995) [36] use the following GDL demand functions to represent the demand side in the MREEM-GDL.

GDL demand

$$q_{itr} = a_{itr} \prod_{n=1}^{t} p_{inr}^{-e_{ir}^{t-n} b_{iir}} \prod_{j \in H_{ir}} \prod_{n=1}^{t} p_{jnr}^{-e_{ir}^{t-n} b_{ijr}} \tag{4.2}$$

where e_{ir} is the lag elasticity of the commodity i: region r (normally. $0 < e_{ir} < 1$).

Therefore the demands are functions of not only current period prices but also previous period prices.

4.2 Decoupling algorithm and PIES method

Fuller and Wu (1989) [12] have shown that it can be difficult to obtain the solution of the MREEM-GDL by the PIES method because the PIES convergence condition is usually violated. For this reason. the Decoupling algorithm has been developed for the solution of the MREEM-GDL. At each iteration of the algorithm. the demand functions containing the lags are replaced by functions in which current demands depend only on current prices. That is. demands in different periods are decoupled. The price elasticities of the decoupled demand functions are *inflated* through the decoupling procedure. An equilibrium solution of the decoupled MREEM-GDL can be calculated with the PIES method as an approximate equilibrium solution of the original MREEM-GDL. The approximate equilibrium price is inserted back in the decoupling procedure to dynamically re-adjust the inflated price elasticities. Successive re-estimation of the inflated price elasticities leads to more accurate estimates of equilibrium. as the iterations proceed.

Wu and Fuller (1995) [36] used the following equation to calculate the inflated price elasticities for the decoupling procedure.

$$B_{ijtr} = \{e_{ir}^{t-1}\frac{ln(p^*_{j1r})}{ln(p^*_{jtr})} + e_{ir}^{t-2}\frac{ln(p^*_{j2r})}{ln(p^*_{jtr})} + \ldots + e_{ir}^{1}\frac{ln(p^*_{j(t-1)r})}{ln(p^*_{jtr})} + 1\}b_{ijr} \qquad (4.3)$$

where B_{ijtr} is the inflated price elasticity of demand q_{itr} with respect to the price p_{jtr} of demand q_{jtr} at time t. region r: p^*_{jtr} is the estimated equilibrium price at the latest decoupling step. At each decoupling iteration. the cross-price elasticity (b_{ijr}) in the equation (4.1) is replaced by B_{ijtr} to reduce the GDL demand side to a decoupled demand side.

4.3 The problem statement

The new decomposition method in Chapter 3. based on the Dantzig-Wolfe principle and the Decoupling algorithm. for the MREEM-GDL. fails to converge in all tests. for reasons explained below.

Step 1 and Step 2 of the new decomposition method in Chapter 3 can be directly applied to the MREEM-GDL. However. in Step 3 for the MREEM-GDL. all equilibrium subproblems are to be solved by the Decoupling algorithm. According to our computational experience. we found that the signs of the inflated price elasticities calculated by equation (4.3) keep changing. e.g.. from positive to negative. in solving equilibrium subproblems. Consequently. at some iterations of the Decoupling algorithm. after the first iteration. NLPs in the PIES sequence incorporate demands that increase with their own prices. causing PIES to fail to converge. The convergence condition of the PIES method is violated. We refer to this problem as the sign-change problem. Another new decomposition method is developed in order to resolve this sign-change difficulty.

4.4 The GDL-decomposition method, for MREEM-GDL

Another new decomposition method. named GDL-decomposition method. is developed for the MREEM-GDL. We noticed that the sign-change problem does not occur in the master equilibrium problem. Therefore. in the GDL-decomposition method. the inflated price elasticities of the equilibrium subproblems are determined by the equilibrium master problem. That is. the equilibrium master problem passes not only the dual price but also the fixed inflated price elasticities to the equilibrium subproblems. Such fixed inflated price elasticities change the demand side of the equilibrium subproblems from GDL-demand to demand depending only on current period prices. Such submodels can be solved by the PIES method. The Decoupling algorithm is not required in solving the equilibrium subproblems. The sign change problem is therefore solved.

The GDL-decomposition method. based on the Dantzig-Wolfe decomposition principle. is composed of three main steps similar to the new decomposition method in Chapter 3. The first two steps are the same as the new decomposition in Chapter 3. The third step includes passing of the inflated price elasticities and the dual prices of the linking constraints from the master problem to the subproblems.

Step 1: Decompose the supply side of the MREEM-GDL into a master problem and regional supply subproblems. defined from the corresponding regional objective function. demand requirement constraints (2.10) and supply constraints (2.12). by the Dantzig-Wolfe decomposition principle.

Step 2: Attach the regional GDL-demand side. $q_r = Q_r(p_r)$. to the corresponding regional supply subproblems and the master problem such that all supply subproblems and the master problem are transformed into equilibrium subproblems and equilibrium master problem. .

Step 3: Solve the decomposed MREEM-GDL by passing the dual price information from the master linking constraints (2.11). and the inflated price elasticities (B_{ijtr}) from the master problem to the subproblems. and quantity information from all equilibrium subproblems to the master problem in an iterative manner until the equilibrium is found.

At each iteration of the GDL-decomposition method. we have the following decomposition-based information exchange. All equilibrium subproblems are solved by the PIES method. with a given estimate of the price vector β and the corresponding inflated price elasticities to provide a new equilibrium proposal. The restricted master problem. solved by the Decoupling algorithm. estimates a new dual price vector β and the inflated price elasticities based on all the accumulated proposals. This iterative process will be terminated by a stopping condition. as discussed in the next section.

4.4.1 Stopping and convergence conditions

In Chapter 3. we mentioned that the convergence properties are not as easy to investigate as for the Dantzig-Wolfe decomposition principle. because the feasible region of the proposals generated by the equilibrium subproblems changes with q_r in the iterations. We encounter the same difficulties for the algorithm for the MREEM-GDL. and in addition. we face the lack of a general convergence proof for the Decoupling algorithm. However. the stopping conditions from Chapter 3 still apply. and we are assured that when the algorithm stops. it has found the equilibrium. We next define the algorithm for the MREEM-GDL. and then present some empirical results.

4.4.2 The algorithm, with the Dantzig-Wolfe principle

Based on the GDL-decomposition method with the adopted Dantzig-Wolfe decomposition principle. we have the following algorithm for solving the MREEM-GDL. Note that the sufficient and the necessary stopping conditions are employed here in the same way as in the MREEM. The algorithm begins by solving each subproblem by the Decoupling algorithm. because we never encountered the sign change problem in the first solutions of the subproblem. It is possible to define an alternate Step 1 wheih uses a guess of the inflated price elasticities. and the PIES method. but we did not follow this path.

Step 1: Set $\beta = \beta^0$ a guess provided by modelers: solve all ESP-r by the Decoupling algorithm: index the proposal with $k = 1$.

Step 2: Solve the REMP^k by the Decoupling algorithm. If $k \geq 2$. go to Step 3: else go to Step 4.

Step 3: Check whether to continue. using the necessary stopping condition. If yes. go to Step 4: else go to Step 5.

Step 4: Solve all ESP-r by the PIES method with equilibrium dual price β and the corresponding inflated price elasticities B_{ijtr} from Step 2: increment $k = k+1$: index the proposal with k: go to Step 2.

Step 5: Check if the MREEM-GDL is solved with the sufficient stopping condition. i.e. solve m LPs defined as ESP-r with fixed demand q_r and β provided from Step 2. and apply the Dantzig-Wolfe stopping condition. If yes. stop. Else. go to Step 4.

4.5 Empirical results

In order to investigate the computational behavior of the GDL-decomposition method. a realistic MREEM-GDL. the Canadian Energy Model with GDL demand. labeled CEM-GDL. from Wu and Chung (1997) [35]. is used. The structure of the CEM-GDL is the same as the CEM in Chapter 3. except for the GDL demand and the time horizon (three periods corresponding to a three-year duration for total time span of nine years from 1986-1995). The CEM-GDL solved by the GDL-decomposition method with the Dantzig-Wolfe decomposition principle was. called CEM-GDL-DW. coded into a GAMS program which was executed in an IBM RS/6000 workstation.

The reference method is the Decoupling method without decomposition. The model with the reference method. referred to as CEM-GDL-REF. was coded in order to have reference results for evaluating the performance of the new decomposition method. In the decomposition method. artificial variables with large cost coefficients are added in each linking constraint: upper bounds are imposed on all variables which do not have any upper bounds: these measures are taken to prevent infeasibility or unboundedness caused by decomposition. The first guess of the linking prices. β^0. was the zero vector. for all tests.

The measures "D-ITER". "TIMEUSED". "MAX%DIFF". and "ELAPSED" of the CEM-GDL model and its reference model CEM-REF are reported as follows. Their definitions are the same as the ones in Table 3.1.

As shown in Table 4.1. there is no significant difference in the calculated demands and prices between the new decomposition method and the reference method.

Model	D-ITER	TIMEUSED	MAX%DIFF	ELAPSED
CEM-GDL-DW	41	29.53	0.220	455
CEM-GDL-REF	—	2.44	—	82

Table 4.1: The computational results of the GDL-decomposition method

The maximum difference of any demand quantity or price is only 0.220%. We note that the computation time used by the decomposition method is longer than the reference method. but both times are short enough to be acceptable to modelers. In general. the GDL-decomposition method shares the same computational properties with the original one. but the additional inflated price elasticities are required to pass from the equilibrium master problem to all equilibrium subproblems.

4.6 Summary

With geometric distributed lag GDL demand in the MREEM. the Decoupling algorithm is used to search for the equilibrium. The Decoupling algorithm employs the inflated price elasticities (equation 4.3) to carry out the decoupling step. While applying the new decomposition in Chapter 3 with the Dantzig-Wolfe decomposition principle and the Decoupling algorithm to the model. a divergent result may be obtained. In solving the equilibrium subproblems with the Decoupling algorithm. the sign of the inflated price elasticities keeps changing. e.g.. from positive to negative. With unrealistic signs for price elasticities. the PIES method fails to converge for some iterations of the Decoupling algorithm. Consequently. the original new decomposition method fails to converge for the MREEM-GDL.

To overcome the above difficulty. another new decomposition method. called

GDL-decomposition method. has been developed. In the GDL-decomposition method. the inflated price elasticities are calculated only by the equilibrium master problem. Such inflated price elasticities are then passed. with the dual prices. from the equilibrium master problem to the corresponding regional equilibrium subproblems. The equilibrium subproblems no longer need to calculate the inflated price elasticities. which can be solved by the PIES method. A test on a realistic model indicated great promise to use the GDL-decomposition method for the MREEM-GDL.

Further research work may concentrate on the possibility of adopting other decomposition principles and the corresponding convergence properties.

Chapter 5

A New Demand-Supply Decomposition Method

In the absence of appropriate demand-supply decomposition methods. some researchers (see Murphy et al. (1988) [23]. Murphy and Mudrageda (1998) [22]. Wagner (1980) [34]. Mansurand Whalley (1982) [19]. and Bueler (1997) [7]) adopted the cobweb algorithm into their tailor-made demand-supply decomposition methods. By means of the cobweb algorithm. prices and quantities are to be passed between the supply and demand sides as a way to decompose an economic equilibrium model. Unfortunately. as can be shown by examining even a one-dimensional example. the cobweb algorithm may diverge in some cases. We refer to this type of decomposition method as the cobweb-decomposition method.

This chapter presents a new demand-supply decomposition method. based on the Dantzig-Wolfe decomposition principle [9]. for solving economic equilibrium models (non-optimization methods). The new decomposition method inherits the finite convergence property of Dantzig-Wolfe decomposition – it must converge in a finite number of iterations. In short. the new decomposition method is created to eliminate the non-convergence shortcoming of existing cobweb-decomposition methods. and also the restriction of the optimization assumption for the EEM.

5.1 The Model

An economic equilibrium model can be written in the following formulation based on the model presented by Ahn and Hogan (1982) [2]. in which the supply side is represented by a detailed cost-minimizing. linear process model. and the demand side by a smooth vector-valued function of prices.

DEM-SUP

Supply side:

$$\min_{x_d, x_s} \quad c_d^T x_d + c_s^T x_s$$

$$s.t. \quad A x_d \geq q \qquad (v) \tag{5.1}$$

$$B_d x_d \leq b_d \tag{5.2}$$

$$B_s x_s \leq b_s \tag{5.3}$$

$$L_d x_d + L_s x_s \leq h \qquad (\beta) \tag{5.4}$$

$$x_d, x_s, q \geq 0$$

Demand side: $q = Q(p)$

Equilibrium Condition: $p = v$

where

c_d = a cost vector for the demand activities:

c_s = a cost vector for the supply activities:

x_d = a demand activity vector:

x_s = a supply activity vector:

$Ax_d \geq q$ represents demand requirement constraints:

v = an optimal dual variable vector (shadow price vector) corresponding to the demand requirement constraints $Ax_d \geq q$:

$B_d x_d \leq b_d$ represents demand activity constraints:

$B_s x_s \leq b_s$ represents supply activity constraints:

$L_d x_d + L_s x_s \leq h$ represents linking demand-supply activity constraints:

β = an optimal dual price vector for the linking demand-supply activity constraints $L_d x_d + L_s x_s \leq h$:

$Q(p)$ = a vector-valued market demand function defined over prices. p.

The demand variable vector q consists of two or more commodity components. The vectors c_d. c_s. x_s. x_d. b_s. b_d. h. p. and matrices L_s. L_d. B_s. B_d have appropriate dimensions.

It is worth mentioning here that the dimensions of B_s are usually huge in large-scale DEM-SUP compared with the size of other matrices. The new decomposition method takes B_s out of the master problem of the DEM-SUP and puts it into the subproblem representing the supply side. which is just a linear program and is therefore easier to solve than an equilibrium model.

5.2 Demand-supply decomposition method

A new demand-supply decomposition method is developed. in which the DEM-SUP is to be decomposed into an equilibrium master problem and a linear programming supply subproblem based on the Dantzig-Wolfe decomposition principle. The PIES method (or any other equilibrium seeking algorithm) and any linear programming algorithm are used to solve the decomposed DEM-SUP.

There are three main steps in the new decomposition method.

Step 1: Decompose the supply side of the DEM-SUP into a **supply subproblem** with all supply activity constraints (5.3) and a master problem with demand requirement constraints (5.1). demand activity constraints (5.2) and linking constraints (5.4). by the Dantzig-Wolfe decomposition principle.

Step 2: Attach the demand side to the master problem such that the master problem is transformed into an **equilibrium master problem.**

Step 3: Solve the decomposed DEM-SUP by exchanging the dual price information β from the linking constraints (5.4) of the **equilibrium master problem** and the accumulated quantity information (called proposals) from the

supply sub-problem in an iterative manner until the equilibrium point is found.

5.2.1 Further decomposition and applications

Although our method explicitly decomposes the DEM-SUP into one **equilibrium master problem** and one **supply subproblem**. the supply subproblem can be further decomposed into more subproblems according to the modeling attributes. such as commodity or region.

An example application for further decomposition in the supply side can be found in economic equilibrium models for energy-environmental policy analyses. Existing real-world models. e.g.. the National Energy Modeling System [22] and the Intermediate Future Forecasting System [23] for the U.S. Energy Information Administration. always have extensive regional detail and allow for the modeling of the various fuel sectors individually in the supply side. For example. if the supply side can be divided into m regions. one can apply the new demand-supply decomposition method such that the **subscript** s in the DEM-SUP can be replaced by an **index** r **to represent regions** and therefore we will have m separable supply subproblems with regional detail and one equilibrium master problem for searching for the equilibrium solution.

Consequently. we can use the concept of a central integrating approach that searches for an equilibrium solution for all regional supply simultaneously without the cumbersome burden of a huge number of supply variables and constraints. That is. the new method can provide a reliable way to manage a complex model by assigning one person responsibility for each regional supply subproblem or the equilibrium master problem. The model is then partitioned and therefore modeling difficulty can be reduced greatly.

Similarly. one can apply the new demand-supply decomposition method to further decompose the supply side by commodity. For example. the supply side of an energy-equilibrium model can be decomposed into several supply subproblems representing different fuel types such as electricity. gas. and oil.

Obviously. this new method can also be used in the DEM-SUP with the GDL demand side. The equilibrium master problem is to be solved by the Decoupling algorithm and the LP supply subproblems are to be solved by any LP solution method. Hence. the sign-change problem can be avoided. because the Decoupling algorithm is not used in the subproblems.

5.3 An illustration, with further decomposition

To illustrate the new decomposition method for the DEM-SUP with more than one supply subproblem. the following DEM-SUPm is used.

DEM-SUPm

Supply side:

$$\min_{x_r, x_d} \; c_d^T x_d + \sum_{r=1}^{m} c_r^T x_r$$

$$s.t. \;\; Ax_d \geq q \quad (v)$$

$$B_d x_d \leq b_d$$

$$\begin{array}{lr}
L_d x_d + L_1 x_1 + L_2 x_2 + \ldots + L_m x_m \leq h & (\beta) \\
B_1 x_1 \qquad\qquad\qquad\qquad \leq b_1 & \\
\qquad B_2 x_2 \qquad\qquad\quad \leq b_2 & \\
\qquad\qquad \ddots \quad \vdots \qquad \vdots & \\
\qquad\qquad\qquad B_m x_m \leq b_m & \\
x_d, x_r, q \geq 0 \qquad \forall r &
\end{array}$$

Demand side: $q = Q(p)$

Equilibrium Condition: $p = v$

5.3.1 Illustration of Step 1

Applying the Dantzig-Wolfe decomposition principle to the LP supply side. i.e. for fixed q. we have one restricted master problem and m supply sub-problems in the supply side at the k^{th} iteration.

Supply side at the k^{th} iteration:

Restricted master problem

$$\min_{\lambda_r^k, x_d} c_d^T x_d + \sum_{r=1}^{m} c_r^T X_r^k \lambda_r^k$$

$$s.t. \qquad A x_d \geq q \qquad (v)$$

$$B_d x_d \leq b_d$$

$$L_d x_d + \sum_{r=1}^{m} L_r X_r^k \lambda_r^k \leq h \qquad (\beta)$$

$$e^k \lambda_r^k = 1 \qquad \forall r$$

$$x_d. \lambda_r^k. q \geq 0 \qquad \forall r$$

where

$X_r^k = (x_r^1. x_r^2. \ldots. x_r^k) \qquad \forall r;$

x_r^i = supply subproblem solution from iteration i;

$\lambda_r^k = (\lambda_{r1}. \lambda_{r2}. \ldots. \lambda_{rk})^T$;

$e^k = (1. 1. \ldots. 1)$ unit k-vector.

The r^{th} *Supply subproblems* (**SSP-r**)

$$\min_{x_r} \ (c_r - \beta L_r)^T x_r$$

$$s.t. \ B_r x_r \leq b_r; \ x_r \geq 0.$$

5.3.2 Illustration of Step 2

If we were to extend the above restricted master problem to an equilibrium model. we would need to include the following conditions.

Demand side: $q = Q(p)$. and

Equilibrium Condition: $p = v$.

We define the following equilibrium master problem in a restricted form.

Restricted equilibrium master problem at the k^{th} *iteration* (**REMP**^{k}):

Supply side:

$$\min_{\lambda_r^k, x_d} c_d^T x_d + \sum_{r=1}^{m} c_r^T X_r^k \lambda_r^k$$

$$s.t. \qquad A x_d \geq q \qquad (v)$$

$$B_d x_d \leq b_d$$

$$L_d x_d + \sum_{r=1}^{m} L_r X_r^k \lambda_r^k \leq h \qquad (\beta)$$

$$e^k \lambda_r^k = 1 \qquad \forall r$$

$$x_d, \lambda_r^k, q \geq 0 \qquad \forall r$$

Demand side: $q = Q(p)$

Equilibrium Condition: $p = v$

Since any proposals from supply subproblems satisfy $B_r x_r \leq b_r$, we can leave these constraints out of the equilibrium master problem. Consequently, we have one restricted equilibrium master problem REMP^k and m linear programming supply sub-problem SSP-r. The equilibrium master problem has few constraints, compared to the whole model because the dimensions of q_r, b_d and h are small for the models that we consider.

5.3.3 Illustration of Step 3

According to the Dantzig-Wolfe decomposition principle, we have the following decomposition-based information exchange. At each iteration, the equilibrium master problem is solved by the PIES algorithm (or another algorithm) and all subproblems by any linear programming solution method. With an estimate of the equilibrium dual price vector β from the equilibrium master problem. all supply subproblems are solved to provide a new equilibrium proposal. The restricted equilibrium master problem is solved to estimate a new price vector β with all the accumulated proposals. This iterative process will be terminated by a stopping conditions as discussed next.

5.4 Stopping and convergence conditions

According to the definition of equilibrium condition provided in [2]. if the optimal dual variable v corresponding to the demand requirement constraint $Ax_d \geq q$ in the cost minimizing linear programming supply side is equal to p such that $q = Q(p)$. and $Q(p)$ is strictly monotone. then this is the unique solution to the DEM-SUPm.

5.4.1 Stopping condition

By using the above definition. we can check. at the k^{th} iteration. whether the DEM-SUPm reaches equilibrium through investigating the cost minimizing LP supply side in the DEM-SUPm. and therefore we have the following theoretical sufficient stopping condition:

Stopping Condition

Given $\hat{q}$ from the RMEP^k. if the cost of the LP supply side of the DEM-SUPm is minimized and the corresponding v is equal to p such that $q = Q(p)$. the DEM-SUPm is solved with the unique equilibrium solution.

However. directly solving the full LP supply side of the DEM-SUPm is not consistent with our aim to decompose by demand and supply side. The stopping condition mentioned in [9] can be used to overcome this difficulty. We can investigate the supply subproblem(s) rather than the equilibrium master problem to check that there are no more proposals that can reduce the cost of the equilibrium master problem. That is. we can use the supply subproblem(s) to check if the stopping condition is satisfied and therefore the DEM-SUPm is solved.

5.4.2 Convergence condition

The new demand-supply decomposition method terminates in a finite number of iterations. yielding a solution of the DEM-SUPm. for the following reasons.

The method is a direct implementation of the column generation scheme. At each iteration the supply subproblem provides an extreme point of the supply subproblem's feasible region. x^k. as a proposal to the equilibrium master problem. Because the supply subproblem is a linear programming model which consists of a finite number of extreme points. the method converges in a finite number of iterations. According to Bazaraa (1990) [5]. it is worth mentioning that the method converges provided that a cycling prevention rule is used in both the calculation of master problem and supply subproblem in the presence of degeneracy.

5.5 The algorithm, with the Dantzig-Wolfe principle

When the new decomposition method adopts the Dantzig-Wolfe decomposition principle. we may have the following algorithm for solving the decomposed [DEM-SUPm]. as illustration.

Step 1: Set $\beta = \beta^0$ a guess supplied by the modeler: solve all [SSP-r]: index the proposal with k=1. solve the $[REMP]^k$.

Step 2: Solve all SSP-r with β provided by $[REMP]^k$:

Step 3: Check if the DEM-SUPm is solved by the stopping condition. If yes. stop. Else. set $k = k + 1$: index the proposal with k: solve the $[REMP]^k$ and then go to Step 2.

5.6 Empirical results

In order to investigate the computational behavior of the new demand-supply decomposition method. a realistic 2-region energy equilibrium model. the Canadian Energy Model CEM [35] was solved. The Canadian Energy Model. consists of 14 q-variables. 78 x_s-variables. 55 x_d-variables. and 120 constraints (including 6 linking constraints for demand and supply model). per period. Each of the two periods corresponds to a three-year duration. for a total time span of six years from 1986 to 1992. Based on the CEM. we developed four test models. called CEM-1. CEM-2. CEM-1-β and CEM-2-β. The CEM-1 and CEM-2 consist of one and two supply subproblems respectively and start with the first guess of the dual price equal to zero ($\beta^0 = 0$). Instead of using $\beta^0 = 0$. the CEM-1-β and CEM-2-β start with a first guess of the dual price vector which is assumed in practice to be provided by modelers.

The new method was coded into GAMS programs for the test models which are executable in an IBM RS/6000 work station. We use GAMS for the coding language because it can provide access to the nonlinear programming solver MINOS 5.3 from a procedural language. The reference method is the PIES method for our empirical tests. solving the original model directly with no decomposition method. The model with the reference method. CEM-REF in Chapter 4. was used in order to have reference results for evaluating the performance of the new decomposition method.

Supply subproblems are solved by a GAMS built-in linear programming solver. and the equilibrium master problem and the reference model are solved by the PIES method with the same convergence tolerance setting as for the reference method. In the decomposition method. artificial variables with large cost coefficient are added in each linking constraint: upper bounds are imposed on all variables which do not have any upper bounds: these measures are taken to prevent infeasibility or unboundedness caused by decomposition.

We report the "D-ITER". "TIMEUSED". "MAX%DIFF". and "ELAPSED" of the test models. CEM-1. CEM-2. CEM-1-β. CEM-2-β and CEM-REF as follows.

Table 5.1 presents the results from the test model and the reference. We found that the maximum difference between the new method and the reference method

Model	D-ITER	TIMEUSED	MAX%DIFF	ELAPSED
CEM-1	55	11.79	0.52	594
CEM-2	36	9.82	0.552	393
CEM-1-β	49	10.6	0.52	538
CEM-2-β	28	7.41	0.552	312
CEM-REF	—	1.16	—	62

Table 5.1: The computational results of the new demand-supply decomposition method

is acceptable. and the time used by the new method is longer than the reference method.

Results concerning the accuracy of the new decomposition method are very encouraging. Although. from Table 5.1. we note that the solution time of the new decomposition method is slower than with reference models. the solution time and the elapsed time is so short (compared with the modeling time) that it is not a great concern to modelers. The elapsed time for decomposition time could be reduced greatly if the whole procedure were coded in a language such as C or C++. GAMS spends a lot of time generating each NLP in the equilibrium calculations for the master problem. and generating each LP for the subproblems.

With the Dantzig-Wolfe decomposition principle for linear programming. Dirickx and Jennergren (1979) [10] mentioned that the number of decomposition iterations decreases with an increase of the number of subproblems. Results from Table 5.1 show that the number of decomposition iterations for the test models decreases from 55 for the model with one subproblem CEM- 1 to 36 for the model with two subproblem CEM-2. and from 49 to 28 when a first guess of β is provided. We believe that the new decomposition method for the DEM-SUPm shares the same relationship between the number of decomposition iterations and subproblems.

Concerning a good first guess of the dual price. the results also show that a good first guess provided by the modeler can decrease the number of decomposition iterations. Check the column **D-ITER**: the number of iterations for both CEM-1 and CEM-2 is 6 or 8 greater than that for CEM-1-β and CEM-2-β respectively. This suggests that if the modeler wants to reduce the decomposition iterations. he should provide a good first guess of the dual price for the linking constraints.

5.7 Summary

In this Chapter. a new demand-supply decomposition method has been developed for modeling and solving asymmetric economic equilibrium models (non-optimization problems). Existing decomposition methods for linear or nonlinear programming can be applied only to decompose symmetric economic equilibrium models (optimization problems). However. real-world economic equilibrium models are usually asymmetric (non-optimization problems). In the absence of suitable decomposition methods. some researchers used the cobweb-decomposition method as a decomposition tool for real-world asymmetric models. However. cobweb-decomposition methods may fail to converge. The new demand-supply decomposition method overcomes the non-convergence shortcoming of the existing demand-supply decomposition method. By adopting the Dantzig-Wolfe decomposition principle. the new demand-supply decomposition method can decompose an asymmetric equilibrium model into one equilibrium master problem and many LP supply subproblems. Since the supply subproblems are linear programmes consisting of a finite number of extreme points. the new demand-supply decomposition method terminates in a finite number of iterations. The possibility of adopting other decomposition principles can be a further research topic.

Chapter 6

Conclusion and Further Research

6.1 Summary

This dissertation explores how to use existing decomposition principles of Linear Programming (LP) to integrate several regional economic equilibrium models (EEMs) as a whole multi-regional economic equilibrium model (MREEM). If the MREEM can be converted into an optimization model under the often unrealistic assumption of symmetric demand. existing LP or NLP decomposition principles can be directly applied for the integration purpose. On the other hand. before now there has been no appropriate decomposition method for the MREEM like LP decomposition principles for optimization models. This dissertation resolves this difficulty by developing new decomposition methods for the non-optimization MREEM in order to alleviate the difficulties of model development and maintenance.

The new decomposition methods have been developed for economic equilibrium models with application to decomposition by region. In general. the new decomposition methods take advantage of the special structure of the supply side of the MREEM. The supply side with special structure. e.g.. block-angular structure. is first decomposed by existing LP decomposition principles accordingly into regional supply subproblems (and a master problem. depending upon the adapted decomposition principles). The corresponding demand functions are then attached to the decomposed supply subproblems (and the master problem) in order to form the decomposed equilibrium subproblems (and the master problem). By means of decomposition-based information exchange among decomposed regional models. the original MREEM can be solved.

Based on the above general approach. three new decomposition methods were developed. The first new decomposition method. presented in Chapter 3. was created for the integration of all regional (existing or new) EEMs. The second one was developed because the first new decomposition method diverges in solving the MREEM-GDL which is a MREEM with GDL structure in order to consider the time-lagged effect in the demand side. The third one is a new demand-supply decomposition method. and is motivated by the non-convergence of the existing cobweb-decomposition methods. The new decomposition methods can be applied in many different areas. e.g.. to study regional tax policy. international trade. or issues in energy economics such as the cost of carbon dioxide emission permits. for example.

Preliminary empirical results have indicated great promise to use the new decomposition methods for the integration purposes. in which regional economic equilibrium models can be modeled individually and then integrated as a multi-regional model. According to all empirical results. there are no significant differences in equilibrium solutions between the new decomposition methods and the reference methods. The maximum difference is only 0.552% over all demand quantities and prices. and over all tests of all algorithms. The computation time of the new decomposition methods is longer than for the reference methods. but the solution time is acceptably short. considering expected reductions in the modeling time. Furthermore. as Murphy and Mudrageda (1998) mentioned. the solution time is not a serious issue because the submodels can be distributed over workstations that could be run in parallel.

6.2 Contribution

Although LP decomposition algorithms can make large-scale linear and nonlinear programming models more manageable. economic equilibrium modelers cannot always use these techniques because many equilibrium models cannot be converted into optimization problems. This research contributes new decomposition methods for non-optimization economic equilibrium models.

The new decomposition methods provide a general approach to combine the solution methods of economic equilibrium models and the LP decomposition principles. As a result. **modelers can integrate regional non-optimization EEMs and/or pure optimization EEMs as a whole multi-regional model.** Without the new decomposition methods. only pure optimization EEMs can be integrated.

Based on the decomposition-based information exchange approach in the new methods. individual complex regional EEMs can be developed. debugged. and implemented in different workstations. Without the new decomposition methods. the non-optimization EEMs cannot be debugged individually. All the significant debugging tasks of the regional EEMs and the linking procedures must be done in the context of the integrated MREEM. Hence. the new decomposition methods provide a traceable modeling environment in which the regional non-optimization EEMs can be well developed before starting the integrating procedures. Modelers may have error-free regional EEMs and therefore they can concentrate only on the integrating procedures in developing a MREEM.

On the other hand. the size of the MREEM increases with the number of regional models. Therefore. it can happen that the MREEM is too huge to be implemented in a single computer. and/or solved by a single commercial solver. Because the new methods enable the smaller size regional models to be implemented in individual workstations. the requirement of the huge size workstation can be eliminated. Before now. such an implementation was infeasible for non-optimization EEMs.

In the realm of theory. the new decomposition methods provide a new decomposition approach. mathematically. for solving asymmetric equilibrium models (e.g. asymmetric variational inequality problems) which consist of a special structure. i.e. block-angular linear constraint set. with hard and easy variables. Without the new decomposition methods. LP/NLP decomposition principles can be applied only on optimization models. If we consider that LP/NLP models are a special case of equilibrium models. this dissertation provides a new way that the application area of existing LP/NLP decomposition principles can be extended to equilibrium models.

6.3 Further Research

6.3.1 Computational Aspect

Different decomposition principles can be adopted by the new decomposition methods. which lead to different versions of the algorithm. More investigation can focus on the possibility of adopting other decomposition principles. For example. Nurminski and Balabanov (1983) [25] applied a primal-dual decomposition method to a large-scale LP energy model which is naturally divided into a supply side and a demand side. We may study if such LP decomposition method can be adapted into the new decomposition methods for solving not only LP energy models but also EEM.

Although the main objective of the new decomposition methods is to reduce the difficulty of the modeling task and the modeling time. efficient implementation of the new decomposition method. e.g.. parallel implementation. is another important issue in order to improve the computation speed.

The new decomposition methods provide an environment in which regional models can be solved individually in their own workstations in order to solve the multi-regional model. One of the possible research areas is how to use the Internet to connect the existing regional EEMs and compute the integrated MREEM efficiently.

6.3.2 Convergence Aspect

We have proven that the new demand-supply decomposition method converges. since its convergence properties are the same as that of the Dantzig-Wolfe decomposition principle because the supply supply subproblems are LP.

However. the convergence properties of the other new decomposition methods are not as easy to investigate as the LP decomposition principles. The convergence theorem of the LP decomposition principles is based on the finite number of extreme points generated by subproblems. However. in the other new decomposition methods. the equilibrium subproblems will generate extreme points of the feasible sets. but these sets change because the regional demand q_r changes in the iterations. Further theoretical investigation of convergence is required.

6.3.3 Application Aspect

LP decomposition principles have been applied in stochastic programming and mixed integer programming. It is worthwhile to explore the possibility of applying the new decomposition methods in stochastic EEMs and mixed integer EEMs.

Decomposition of the EEM by different modeling attributes (e.g.. time. commodity. and user-group) is another worthwhile research area. because some research groups maintain their models by different attribute-oriented teams. For example. a large-scale energy equilibrium model may be maintained by different teams for different fuel types (oil. gas. electricity). or for different user-groups (users from industries. commerce. transportation). or for different time-periods.

Harker and Pang (1990) [13] showed that the VI problems can be applied in various situations. such as the Nash equilibrium of an n-person non-cooperative game. traffic assignment or network equilibrium model. etc. Since the new decomposition methods can be generalized in the VI form. many more application areas can be explored.

The new decomposition methods rely upon the assumption of the separability of the demand functions. i.e.. the independence of demand from other regions' prices. which restricts the application of our methods to the VI problems. Another restriction is our assumption of a linear supply model. Further research may examine decomposition methods for different forms of cost functions in the supply side. such as nonlinear or stochastic cost functions. and for different kinds of demand functions. such as non-separable demand functions.

Appendix A

GAMS File of Small Two-region Model

Two GAMS files of Small-DW and Small-LF in Chapter 3 are attached in this Appendix. for reference. Other GAMS files regarding the Canadian Energy Model and the Canadian-USA Energy Model are too large (more than 200 pages) to be attached and therefore those files are ommitted for brevity.

A.1 With Dantzig-Wolfe Principle, Small-DW

```
**[Two-region model] solved by the new decomposition method with
**Dantzig-Wolfe decomposition principle.
set k /1*40/;
set km /1*40/;
SET     ITER    /1*50/;
SETS I1 plants /PLANT11,PLANT12,PLANT13/
     I2 plants /PLANT21,PLANT22,PLANT23/
     J1 retailing regions /REGION11, REGION12, REGION13/
     J2 retailing regions /REGION21, REGION22, REGION23/;
ALIAS (J1,EJ1);
ALIAS (J2,EJ2);
set k1(k);
set k2(k);
alias(kloop,k);
alias(kloop1,k);
alias(kloop2,k);
k1('1')=yes;
k2('1')=yes;
parameters pi11,pi12,pi21,pi22;
PARAMETERS
GAMMA1(I1)  supply price intercepts
/PLANT11 42
 PLANT12 35
 PLANT13 50 /
DELTA1(I1)   supply price coefficients
/PLANT11 0.30
                        PLANT12   0.25
                        PLANT13   0.50 /
    GAMMA2(I2)  supply price intercepts
                       /PLANT21   40
                        PLANT22   30
                        PLANT23   45 /
    DELTA2(I2)   supply price coefficients
                       /PLANT21   0.2
                        PLANT22   0.1
                        PLANT23   0.3 /
    EA1(J1) demand fct. intercepts (elasticities)
/REGION11   350
 REGION12   475
 REGION13   300/
    EA2(J2) demand fct. intercepts (elasticities)
/REGION21   325
 REGION22   425
 REGION23   300/;

TABLE   EB1(J1,EJ1)   elasticities constant
REGION11 REGION12 REGION13
REGION11 -2 0.1 0.2
```

```
REGION12 0.2 -2.5 0.05
REGION13 0.1 0.15 -1.7 ;
TABLE   EB2(J2,EJ2)   elasticities constant
REGION21 REGION22 REGION23
REGION21 -2 0.1 0.2
REGION22 0.2 -2.5 0.05
REGION23 0.1 0.15 -1.7 ;

*******************************************************

TABLE
  C1(I1,J1)      unit shipment costs
                        REGION11            REGION12          REGION13
      PLANT11              0                   1                 1.5
      PLANT12              1                   0                 2.0
      PLANT13              1.5                2.0                 0  ;
TABLE
  C2(I2,J2)      unit shipment costs
                        REGION21            REGION22          REGION23
      PLANT21              0                   1                 1.5
      PLANT22              1                   0                 2.0
      PLANT23              1.5                2.0                 0  ;


VARIABLES
   SURPLUS       total surplus
   cost1
   cost2

POSITIVE VARIABLES
   X1(I1,J1) quantity shipped from plant I to region J
   S1(I1)        supply at plant I
   D1(J1) demand at region J
   X2(I2,J2) quantity shipped from plant I to region J
   S2(I2)        supply at plant I
   D2(J2) demand at region J
   lv11 linking variables LV..
   lv12
   lv21
   lv22
   la1(k)
   la2(k)
   slack1
   slack2
   slack3
   slack4
   slack5
   slack6
   slack7
   slack8
;
lv11.up=500;
lv12.up=500;
lv21.up=500;
lv22.up=500;

EQUATIONS
   OBJECTIVE     objective function defined
   SUPBAL1(I1)  commodity balance at each plant I1
   SUPBAL2(I2)  commodity balance at each plant I2
   linksup1(I1)
   linksup2(I2)
   DEMBAL1(J1)  commodity balance at each retailing region J1
   DEMBAL2(J2)  commodity balance at each retailing region J2
   linkdem1(J1)
   linkdem2(J2)
   mDEMBAL1(J1)  commodity balance at each retailing region J1
   mDEMBAL2(J2)  commodity balance at each retailing region J2
   mldem1(J1)
   mldem2(J2)
   link1
   link2
   subobj1
   subobj2
   sumla1
   sumla2
;

PARAMETER P11(J1)
/REGION11 110.6
 REGION12 109.6
 REGION13 111.6/;
```

```
PARAMETER P21(J2)
/REGION21 110.6
 REGION22 109.6
 REGION23 111.6/;
parameter pcost1(k),pcost2(k);
PARAMETER PV1(J1) ******** for PIES ****************;
        PV1(J1) = 200;
PARAMETER PV2(J2);
        PV2(J2) = 200;
parameter mp11(J1),mp21(J2);
mp11(J1) = 200;
mp21(J2) = 200;
parameter plv11(k),plv12(k),plv21(k),plv22(k),pp11(J1),pp21(J2);
parameter px1(I1,J1,k),px2(I2,J2,k),ps1(I1,k),ps2(I2,k),pd1(J1),pd2(J2);
parameter md1(J1),md2(J2);
parameter timemas(k,ITER),timesm1(k,ITER),timesm2(k,ITER),
  itermas(k,ITER),itersm1(k,ITER),itersm2(k,ITER);

pp11(J1)=P11(J1);
pp21(J2)=P21(J2);
OBJECTIVE..
SURPLUS =E= SUM(k2,la1(k2)*pcost1(k2))+sum(k2,la2(k2)*pcost2(k2))
+ 10000*(slack1+slack2+slack3+slack4+slack5+slack6+slack7+slack8)
-SUM(J1,(EA1(J1)+SUM(EJ1$(ORD(EJ1) NE ORD(J1)),EB1(J1,EJ1)*mp11(EJ1)))
/(-EB1(J1,J1))*D1(J1)-D1(J1)*D1(J1)/(2*(-EB1(J1,J1))))
-SUM(J2,(EA2(J2)+SUM(EJ2$(ORD(EJ2) NE ORD(J2)),EB2(J2,EJ2)*mp21(EJ2)))
/(-EB2(J2,J2))*D2(J2)-D2(J2)*D2(J2)/(2*(-EB2(J2,J2))))
;
link1..  -sum(k2,plv11(k2)*la1(k2))+sum(k2,plv12(k2)*la2(k2))
  -(slack1)+slack7=e=0;
link2..  -sum(k2,plv21(k2)*la2(k2))+sum(k2,plv22(k2)*la1(k2))
          -(slack2)+slack8=e=0;
sumla1..  sum(k2,la1(k2))=e=1;
sumla2..  sum(k2,la2(k2))=e=1;
mDEMBAL1(J1)$(ord(J1) ne 3)..
SUM(I1,sum(k2,px1(I1,J1,k2)*la1(k2)))+slack3=g=D1(J1);
mDEMBAL2(J2)$(ord(J2) ne 3)..
SUM(I2,sum(k2,px2(I2,J2,k2)*la2(k2)))+slack4=g=D2(J2);
mldem1(J1)$(ord(J1) eq 3)..
SUM(I1,sum(k2,px1(I1,J1,k2)*la1(k2)))+sum(k2,plv22(k2)*la1(k2))+slack5
=g= D1(J1);
```

```
mldem2(J2)$(ord(J2) eq 3)..
SUM(I2,sum(k2,px2(I2,J2,k2)*la2(k2)))+sum(k2,plv12(k2)*la2(k2))+slack6
=g= D2(J2);

subobj1.. cost1=e=
SUM(J1,(EA1(J1)+SUM(EJ1$(ORD(EJ1) NE ORD(J1)),EB1(J1,EJ1)*P11(EJ1)))
/(-EB1(J1,J1))*D1(J1)-D1(J1)*D1(J1)/(2*(-EB1(J1,J1))))
        -SUM(I1,GAMMA1(I1)*S1(I1))
- SUM((I1,J1),C1(I1,J1)*X1(I1,J1))-0.25*lv22-0.25*lv11
+(pi11*(-lv11)+pi21*(lv22))
;
subobj2.. cost2=e=
SUM(J2,(EA2(J2)+SUM(EJ2$(ORD(EJ2) NE ORD(J2)),EB2(J2,EJ2)*P21(EJ2)))
/(-EB2(J2,J2))*D2(J2)-D2(J2)*D2(J2)/(2*(-EB2(J2,J2))))
        -SUM(I2,GAMMA2(I2)*S2(I2))
- SUM((I2,J2),C2(I2,J2)*X2(I2,J2))-0.25*lv12-0.25*lv21
+(pi11*(lv12)+pi21*(-lv21))
;

SUPBAL1(I1)$(ord(I1) ne 2)..   SUM(J1,X1(I1,J1)) - S1(I1) =L= 0;
SUPBAL2(I2)$(ord(I2) ne 2)..   SUM(J2,X2(I2,J2)) - S2(I2) =L= 0;
linksup1(I1)$(ord(I1) eq 2)..   SUM(J1,X1(I1,J1)) + lv11 - S1(I1) =L= 0;
linksup2(I2)$(ord(I2) eq 2)..   SUM(J2,X2(I2,J2)) + lv21 - S2(I2) =L= 0;
DEMBAL1(J1)$(ord(J1) ne 3)..   SUM(I1,X1(I1,J1)) =g= D1(J1);
DEMBAL2(J2)$(ord(J2) ne 3)..   SUM(I2,X2(I2,J2)) =g= D2(J2);
linkdem1(J1)$(ord(J1) eq 3)..   SUM(I1,X1(I1,J1)) + lv22 =g= D1(J1);
linkdem2(J2)$(ord(J2) eq 3)..   SUM(I2,X2(I2,J2)) + lv12 =g= D2(J2);

******************************** new
SCALAR  MERR;
SCALAR  ERR1;
SCALAR  ERR2;
pi11=0;
pi21=0;
MODEL subm1/subobj1,SUPBAL1,DEMBAL1,linksup1,linkdem1/;
MODEL subm2/subobj2,SUPBAL2,DEMBAL2,linksup2,linkdem2/;
model master/OBJECTIVE,link1,link2,sumla1,sumla2,
             mDEMBAL1,mDEMBAL2,mldem1,mldem2/;

MERR=0.2;
LOOP (ITER,
   IF (MERR GT 0.001,
       SOLVE subm1 USING NLP MAXIMIZING cost1;
       PV1(J1) = (EA1(J1)+SUM(EJ1$(ORD(EJ1) NE ORD(J1)),EB1(J1,EJ1)*P11(EJ1))
                -D1.L(J1))/(-EB1(J1,J1));
       MERR = 0;
       LOOP(J1,
            ERR1 = ABS(PV1(J1)-P11(J1))/P11(J1);
            IF (ERR1 GT MERR,
MERR = ERR1;
               );
            );
       P11(J1) = PV1(J1);
      );
     );

MERR=0.2;
LOOP (ITER,
   IF (MERR GT 0.001,
       SOLVE subm2 USING NLP MAXIMIZING cost2;
       PV2(J2) = (EA2(J2)+SUM(EJ2$(ORD(EJ2) NE ORD(J2)),EB2(J2,EJ2)*P21(EJ2))
                -D2.L(J2))/(-EB2(J2,J2));
       MERR = 0;
       LOOP(J2,
            ERR2 = ABS(PV2(J2)-P21(J2))/P21(J2);
            IF (ERR2 GT MERR,
                MERR = ERR2;
               );
            );
       P21(J2) = PV2(J2);
      );
     );

scalar merr1,merr2,pobj,perr1,perr2,pmerr1,pmerr2;
pobj=10;
merr1=0.2;
merr2=0.2;
pmerr1=0.2;
pmerr2=0.2;

loop(kloop,
        k1(k)=yes$(ord(k) eq ord(kloop));
        k2(kloop)=yes;
plv11(k1)=lv11.l;
plv12(k1)=lv12.l;
plv21(k1)=lv21.l;
plv22(k1)=lv22.l;
px1(I1,J1,k1)=X1.l(I1,J1);
px2(I2,J2,k1)=X2.l(I2,J2);
ps1(I1,k1)=S1.l(I1);
ps2(I2,k1)=S2.l(I2);
pd1(J1)=D1.l(J1);
pd2(J2)=D2.l(J2);

pcost1(k1)=SUM(I1,GAMMA1(I1)*S1.l(I1))
+ SUM((I1,J1),C1(I1,J1)*X1.l(I1,J1))+0.25*lv22.l+0.25*lv11.l;
pcost2(k1)=SUM(I2,GAMMA2(I2)*S2.l(I2))
+ SUM((I2,J2),C2(I2,J2)*X2.l(I2,J2))+0.25*lv12.l+0.25*lv21.l;

 if(merr2 gt 0.00001,
    MERR=0.2;
     LOOP (ITER,
        IF (MERR GT 0.001,
            SOLVE master USING NLP MINIMIZING SURPLUS;
        timemas(k1,ITER)=master.resusd;
        itermas(k1,ITER)=master.iterusd;

       PV1(J1) = (EA1(J1)+SUM(EJ1$(ORD(EJ1) NE ORD(J1)),EB1(J1,EJ1)*mp11(EJ1))
                -D1.L(J1))/(-EB1(J1,J1));
       PV2(J2) = (EA2(J2)+SUM(EJ2$(ORD(EJ2) NE ORD(J2)),EB2(J2,EJ2)*mp21(EJ2))
                -D2.L(J2))/(-EB2(J2,J2));
       MERR = 0;
         LOOP(J1,
            ERR1 = ABS(PV1(J1)-mp11(J1))/mp11(J1);
            IF (ERR1 GT MERR,
                MERR = ERR1;
               );
              );
          LOOP(J2,
            ERR2 = ABS(PV2(J2)-mp21(J2))/mp21(J2);
            IF (ERR2 GT MERR,
                MERR = ERR2;
               );
              );
mp11(J1) = PV1(J1);
mp21(J2) = PV2(J2);
          );
        );

pi11=link1.m;
pi21=link2.m;

pmerr1=0;
       LOOP(J1,
            perr1 = ABS(pp11(J1)-mp11(J1))/mp11(J1);
            IF (perr1 GT pmerr1,
                pmerr1 = perr1;
               );
            );
       LOOP(J2,
            perr2 = ABS(pp21(J2)-mp21(J2))/mp21(J2);
            IF (perr2 GT pmerr1,
                pmerr1 = perr2;
               );
            );
pp11(J1)=mp11(J1);
pp21(J2)=mp21(J2);
merr2=abs(pobj-SURPLUS.l);
pobj=SURPLUS.l;
md1(J1)=D1.L(J1);
md2(J2)=D2.L(J2);

MERR=0.2;
LOOP (ITER,
   IF (MERR GT 0.001,
       SOLVE subm1 USING NLP MAXIMIZING cost1;
        timesm1(k1,ITER)=subm1.resusd;
        itersm1(k1,ITER)=subm1.iterusd;
       PV1(J1) = (EA1(J1)+SUM(EJ1$(ORD(EJ1) NE ORD(J1)),EB1(J1,EJ1)*P11(EJ1))
                -D1.L(J1))/(-EB1(J1,J1));
       MERR = 0;
       LOOP(J1,
            ERR1 = ABS(PV1(J1)-P11(J1))/P11(J1);
            IF (ERR1 GT MERR,
                MERR = ERR1;
               );
            );
       P11(J1) = PV1(J1);
      );
     );

*P21(J2)=mp21(J2);
MERR=0.2;
LOOP (ITER,
   IF (MERR GT 0.001,
       SOLVE subm2 USING NLP MAXIMIZING cost2;
        timesm2(k1,ITER)=subm2.resusd;
        itersm2(k1,ITER)=subm2.iterusd;

       PV2(J2) = (EA2(J2)+SUM(EJ2$(ORD(EJ2) NE ORD(J2)),EB2(J2,EJ2)*P21(EJ2))
                -D2.L(J2))/(-EB2(J2,J2));
       MERR = 0;
       LOOP(J2,
            ERR2 = ABS(PV2(J2)-P21(J2))/P21(J2);
            IF (ERR2 GT MERR,
                MERR = ERR2;
               );
            );
       P21(J2) = PV2(J2);
      );
     );
    );
);
DISPLAY D1.L, PV1,D2.L,PV2,lv11.l,lv12.l,lv21.l,lv22.l,pcost1,pcost2;
parameter splv11,splv12,splv21,splv22,spx2(I2,J2),spx1(I1,J1);
splv11 = sum(k2,plv11(k2)*la1.l(k2));
splv12 = sum(k2,plv12(k2)*la2.l(k2));
splv21 = sum(k2,plv21(k2)*la2.l(k2));
splv22 = sum(k2,plv22(k2)*la1.l(k2));
spx1(I1,J1) = sum(k2,px1(I1,J1,k2)*la1.l(k2));
spx2(I2,J2) = sum(k2,px2(I2,J2,k2)*la2.l(k2));
display splv11,splv12,splv21,splv22,mp11,mp21,spx1,spx2,pobj,md1,md2;
display timemas,timesm1,timesm2,itermas,itersm1,itersm2;
```

A.2 With Lan-Fuller Principle, Small-LF

```
**[Two-region model] solved by the new decomposition method with
**Lan-Fuller decomposition method.

set k /1*20/;
set km /1*20/;
SET     ITER    /1*50/;
SETS I1 plants /PLANT11,PLANT12,PLANT13/
     I2 plants /PLANT21,PLANT22,PLANT23/
     J1 retailing regions /REGION11, REGION12, REGION13/
     J2 retailing regions /REGION21, REGION22, REGION23/;
ALIAS (J1,EJ1);
ALIAS (J2,EJ2);
set k1(k);
set k2(k);
alias(kloop,k);
alias(kloop1,k);
alias(kloop2,k);
k1('1')=yes;
k2('1')=yes;
parameters pi21(k),pi22(k),pi23(k),pi51(k),pi52(k);
PARAMETERS
GAMMA1(I1)  supply price intercepts
        /PLANT11 42
 PLANT12 35
 PLANT13 50 /
DELTA1(I1)   supply price coefficients
       /PLANT11   0.30
                        PLANT12   0.25
                        PLANT13   0.50 /
    GAMMA2(I2)  supply price intercepts
                       /PLANT21   40
                        PLANT22   30
                        PLANT23   45 /
    DELTA2(I2)   supply price coefficients
                       /PLANT21   0.2
                        PLANT22   0.1
                        PLANT23   0.3 /
    EA1(J1) demand fct. intercepts (elasticities)
```

```
/REGION11   350
 REGION12   475
 REGION13   300/
    EA2(J2) demand fct. intercepts (elasticities)
/REGION21   325
 REGION22   425
 REGION23   300/;

TABLE   EB1(J1,EJ1)   elasticities constant
REGION11 REGION12 REGION13
REGION11 -2 0.1 0.2
REGION12 0.2 -2.5 0.05
REGION13 0.1 0.15 -1.7 ;
TABLE   EB2(J2,EJ2)   elasticities constant
REGION21 REGION22 REGION23
REGION21 -2 0.1 0.2
REGION22 0.2 -2.5 0.05
REGION23 0.1 0.15 -1.7 ;

*******************************************************

TABLE
  C1(I1,J1)      unit shipment costs
                        REGION11           REGION12         REGION13
      PLANT11              0                  1               1.5
      PLANT12              1                  0               2.0
      PLANT13              1.5                2.0              0  ;
TABLE
  C2(I2,J2)      unit shipment costs
                        REGION21           REGION22         REGION23
      PLANT21              0                  1               1.5
      PLANT22              1                  0               2.0
      PLANT23              1.5                2.0              0  ;


VARIABLES
   cost11
   cost1
   cost2
   m
POSITIVE VARIABLES
```

```
   X1(I1,J1) quantity shipped from plant I to region J
   S1(I1)       supply at plant I
   D1(J1) demand at region J
   X2(I2,J2) quantity shipped from plant I to region J
   S2(I2)       supply at plant I
   D2(J2) demand at region J
   lv11 linking variables LV..
   lv12
   lv21
   lv22
   la1(k)
   la2(k)
   slack1
   slack2
   slack3
   slack4
   slack5
   slack6
   slack7
   slack8
;

lv11.up=500;
lv22.up=500;

EQUATIONS
   OBJECTIVE    objective function defined
   SUPBAL1(I1)  commodity balance at each plant I1
   SUPBAL2(I2)  commodity balance at each plant I2
   linksup1(I1)
   linksup2(I2)
   DEMBAL1(J1)  commodity balance at each retailing region J1
   DEMBAL21(J1)  commodity balance at each retailing region J2
   DEMBAL22(J2)  commodity balance at each retailing region J2
   linkdem1(J1)
   lindem21(J1)
   lindem22(J2)
   link1
   link2
   subobj11
   subobj1
   subobj2
   sumla1
   cut(k)
;

PARAMETER P11(J1)
/REGION11 110.6
 REGION12 109.6
 REGION13 111.6/;
PARAMETER P12(J2)
/REGION21 110.6
 REGION22 109.6
 REGION23 111.6/;
PARAMETER P21(J1)
/REGION11 110.6
 REGION12 109.6
 REGION13 111.6/;
PARAMETER P22(J2)
/REGION21 110.6
 REGION22 109.6
 REGION23 111.6/;

parameter pcost1(k),pcost2(k);

PARAMETER PV11(J1),PV12(J2),PV21(J1),PV22(J2);
        PV11(J1) = 200;
        PV12(J2) = 200;
        PV21(J1) = 200;
        PV22(J2) = 200;


parameter plv11(k),plv12(k),plv21(k),plv22(k);
* pp11(J1),pp21(J2);
parameter px1(I1,J1,k),px2(I2,J2,k),ps1(I1,k),ps2(I2,k),pd1(J1),pd2(J2);
parameter timesub1(k,ITER),timesub2(k,ITER),scost1(k),scost2(k);

subobj11.. cost11=e=
SUM(J1,(EA1(J1)+SUM(EJ1$(ORD(EJ1) NE ORD(J1)),EB1(J1,EJ1)*P11(EJ1)))
/(-EB1(J1,J1))*D1(J1)-D1(J1)*D1(J1)/(2*(-EB1(J1,J1))))
        -SUM(I1,GAMMA1(I1)*S1(I1))
- SUM((I1,J1),C1(I1,J1)*X1(I1,J1))-0.25*lv22-0.25*lv11
```

```
;
subobj1.. cost1=e=
SUM(J1,(EA1(J1)+SUM(EJ1$(ORD(EJ1) NE ORD(J1)),EB1(J1,EJ1)*P11(EJ1)))
/(-EB1(J1,J1))*D1(J1)-D1(J1)*D1(J1)/(2*(-EB1(J1,J1))))
+ SUM(J2,(EA2(J2)+SUM(EJ2$(ORD(EJ2) NE ORD(J2)),EB2(J2,EJ2)*P12(EJ2)))
/(-EB2(J2,J2))*D2(J2)-D2(J2)*D2(J2)/(2*(-EB2(J2,J2))))
        -SUM(I1,GAMMA1(I1)*S1(I1))
- SUM((I1,J1),C1(I1,J1)*X1(I1,J1))-0.25*lv22-0.25*lv11
+ m
;
SUPBAL1(I1)$(ord(I1) ne 2)..   SUM(J1,X1(I1,J1)) - S1(I1) =L= 0;
linksup1(I1)$(ord(I1) eq 2)..   SUM(J1,X1(I1,J1)) + lv11 - S1(I1) =L= 0;
DEMBAL1(J1)$(ord(J1) ne 3)..   SUM(I1,X1(I1,J1)) =g= D1(J1);
linkdem1(J1)$(ord(J1) eq 3)..   SUM(I1,X1(I1,J1)) + lv22 =g= D1(J1);
cut(k2)..   m=l= pi21(k2)*D2('REGION21') +
 pi22(k2)*D2('REGION22') +
 pi23(k2)*D2('REGION23')
- pi51(k2)*(-lv11) - pi52(k2)*(lv22);

subobj2.. cost2=e=
SUM(J1,(EA1(J1)+SUM(EJ1$(ORD(EJ1) NE ORD(J1)),EB1(J1,EJ1)*P21(EJ1)))
/(-EB1(J1,J1))*D1(J1)-D1(J1)*D1(J1)/(2*(-EB1(J1,J1))))
+ SUM(J2,(EA2(J2)+SUM(EJ2$(ORD(EJ2) NE ORD(J2)),EB2(J2,EJ2)*P22(EJ2)))
/(-EB2(J2,J2))*D2(J2)-D2(J2)*D2(J2)/(2*(-EB2(J2,J2))))
        -SUM(I2,GAMMA2(I2)*S2(I2))
- SUM((I2,J2),C2(I2,J2)*X2(I2,J2))-0.25*lv12-0.25*lv21
- SUM(k2,la1(k2)*pcost1(k2))
- 10000*(slack1+slack2+slack3+slack4)
;

DEMBAL21(J1)$(ord(J1) ne 3)..
SUM(I1,sum(k2,px1(I1,J1,k2)*la1(k2))) =g= D1(J1);
lindem21(J1)$(ord(J1) eq 3)..
SUM(I1,sum(k2,px1(I1,J1,k2)*la1(k2))) +
    sum(k2,plv22(k2)*la1(k2)) =g= D1(J1);
DEMBAL22(J2)$(ord(J2) ne 3)..   SUM(I2,X2(I2,J2)) =g= D2(J2);
lindem22(J2)$(ord(J2) eq 3)..   SUM(I2,X2(I2,J2)) + lv12 =g= D2(J2);
SUPBAL2(I2)$(ord(I2) ne 2)..   SUM(J2,X2(I2,J2)) - S2(I2) =L= 0;
linksup2(I2)$(ord(I2) eq 2)..   SUM(J2,X2(I2,J2)) + lv21 - S2(I2) =L= 0;
link1..  -sum(k2,plv11(k2)*la1(k2))+lv12-(slack1)+slack2=e=0;
link2..  -lv21+sum(k2,plv22(k2)*la1(k2))-(slack3)+slack4=e=0;
```

```
sumla1..  sum(k2,la1(k2))=e=1;

******************************** new
SCALAR  MERR;
SCALAR  ERR11;
SCALAR  ERR12;
SCALAR  ERR21;
SCALAR  ERR22;

MODEL subm11/subobj11,SUPBAL1,DEMBAL1,linksup1,linkdem1/;
MODEL subm1/subobj1,SUPBAL1,DEMBAL1,linksup1,linkdem1,cut/;
MODEL subm2/subobj2,SUPBAL2,DEMBAL21,lindem21,DEMBAL22,
    lindem22,linksup2,link1,link2,sumla1/;

MERR=0.2;
LOOP (ITER,
   IF (MERR GT 0.001,
       SOLVE subm11 USING NLP MAXIMIZING cost11;
       PV11(J1) = (EA1(J1)+SUM(EJ1$(ORD(EJ1) NE ORD(J1)),EB1(J1,EJ1)*P11(EJ1))
                -D1.L(J1))/(-EB1(J1,J1));
       MERR = 0;
       LOOP(J1,
            ERR11 = ABS(PV11(J1)-P11(J1))/P11(J1);
            IF (ERR11 GT MERR,
MERR = ERR11;
               );
            );
       P11(J1) = PV11(J1);
      );
     );

plv11(k1)=lv11.l;
plv22(k1)=lv22.l;
px1(I1,J1,k1)=X1.l(I1,J1);
ps1(I1,k1)=S1.l(I1);
* ps2(I2,k1)=S2.l(I2);
pd1(J1)=D1.l(J1);
* pd2(J2)=D2.l(J2);

pcost1(k1)=SUM(I1,GAMMA1(I1)*S1.l(I1))
+ SUM((I1,J1),C1(I1,J1)*X1.l(I1,J1))+0.25*lv22.l+0.25*lv11.l;
```

```
MERR=0.2;
LOOP (ITER,
   IF (MERR GT 0.001,
       SOLVE subm2 USING NLP MAXIMIZING cost2;
       PV21(J1) = (EA1(J1)+SUM(EJ1$(ORD(EJ1) NE ORD(J1)),EB1(J1,EJ1)*P21(EJ1))
                -D1.L(J1))/(-EB1(J1,J1));
       PV22(J2) = (EA2(J2)+SUM(EJ2$(ORD(EJ2) NE ORD(J2)),EB2(J2,EJ2)*P22(EJ2))
                -D2.L(J2))/(-EB2(J2,J2));
       MERR = 0;
       LOOP(J1,
            ERR21 = ABS(PV21(J1)-P21(J1))/P21(J1);
            IF (ERR21 GT MERR,
                MERR = ERR21;);
            );
       LOOP(J2,
            ERR22 = ABS(PV22(J2)-P22(J2))/P22(J2);
            IF (ERR22 GT MERR,
                MERR = ERR22;);
            );
       P21(J1) = PV21(J1);
       P22(J2) = PV22(J2);
      );
     );

scalar merr1,merr2,pobj,perr1,perr2,pmerr1,pmerr2;
pobj=10;
merr1=0.2;
merr2=0.2;
pmerr1=0.2;
pmerr2=0.2;

loop(kloop,
        k1(k)=yes$(ord(k) eq ord(kloop));
        k2(kloop)=yes;
pi21(k2)=DEMBAL22.m('REGION21');
pi22(k2)=DEMBAL22.m('REGION22');
pi23(k2)=lindem22.m('REGION23');
pi51(k2)=link1.m;
pi52(k2)=link2.m;
```

```
if(merr2 gt 0.0001,

MERR=0.2;
LOOP (ITER,
   IF (MERR GT 0.001,
       SOLVE subm1 USING NLP MAXIMIZING cost1;
timesub1(k1,ITER)=subm1.resusd;
       PV11(J1) = (EA1(J1)+SUM(EJ1$(ORD(EJ1) NE ORD(J1)),EB1(J1,EJ1)*P11(EJ1))
                -D1.L(J1))/(-EB1(J1,J1));
       PV12(J2) = (EA2(J2)+SUM(EJ2$(ORD(EJ2) NE ORD(J2)),EB2(J2,EJ2)*P12(EJ2))
                -D2.L(J2))/(-EB2(J2,J2));
       MERR = 0;
       LOOP(J1,
            ERR11 = ABS(PV11(J1)-P11(J1))/P11(J1);
            IF (ERR11 GT MERR,
MERR = ERR11;);
            );
       LOOP(J2,
            ERR12 = ABS(PV12(J2)-P12(J2))/P12(J2);
            IF (ERR12 GT MERR,
MERR = ERR12;);
            );
       P11(J1) = PV11(J1);
       P12(J2) = PV12(J2);
      );
     );

plv11(k1)=lv11.l;
plv22(k1)=lv22.l;
px1(I1,J1,k1)=X1.l(I1,J1);
ps1(I1,k1)=S1.l(I1);
* ps2(I2,k1)=S2.l(I2);
pd1(J1)=D1.l(J1);
pd2(J2)=D2.l(J2);

pcost1(k1)=SUM(I1,GAMMA1(I1)*S1.l(I1))
+ SUM((I1,J1),C1(I1,J1)*X1.l(I1,J1))+0.25*lv22.l+0.25*lv11.l;
scost1(k1)=cost1.l;
MERR=0.2;
LOOP (ITER,
   IF (MERR GT 0.001,
       SOLVE subm2 USING NLP MAXIMIZING cost2;
timesub2(k1,ITER)=subm2.resusd;
       PV21(J1) = (EA1(J1)+SUM(EJ1$(ORD(EJ1) NE ORD(J1)),EB1(J1,EJ1)*P21(EJ1))
                -D1.L(J1))/(-EB1(J1,J1));
       PV22(J2) = (EA2(J2)+SUM(EJ2$(ORD(EJ2) NE ORD(J2)),EB2(J2,EJ2)*P22(EJ2))
                -D2.L(J2))/(-EB2(J2,J2));
       MERR = 0;
       LOOP(J1,
            ERR21 = ABS(PV21(J1)-P21(J1))/PV21(J1);
            IF (ERR21 GT MERR,
                MERR = ERR21;);
            );
       LOOP(J2,
            ERR22 = ABS(PV22(J2)-P22(J2))/PV22(J2);
            IF (ERR22 GT MERR,
                MERR = ERR22;);
            );
       P21(J1) = PV21(J1);
       P22(J2) = PV22(J2);
      );
     );
scost2(k1)=cost2.l;
merr2=abs(cost1.l-cost2.l);
    );
);
DISPLAY D1.L, PV11,D2.L,PV21,PV12,PV22,lv11.l,lv12.l,lv21.l,lv22.l,pcost1;
parameter splv11,splv12,splv21,splv22,spx2(I2,J2),spx1(I1,J1);
splv11 = sum(k2,plv11(k2)*la1.l(k2));
splv22 = sum(k2,plv22(k2)*la1.l(k2));
spx1(I1,J1) = sum(k2,px1(I1,J1,k2)*la1.l(k2));
display splv11,splv22,spx1,X2.l,scost1,scost2,timesub1,timesub2;
```

Bibliography

[1] B. H. Ahn. *Computation of Market Equilibrium for Policy Analysis: the Project Independence Evaluation System Approach.* PhD thesis. Department of Engineering-Economic System. Stanford University. 1979.

[2] B.H. Ahn and W.W. Hogan. On Convergence of the PIES Algorithm for Computing Equilibria. *Operations Research.* 30(2):281–300. 1982.

[3] O. Bahn. A. Haurie. S. Kypreos. and J.-P. Vial. A Decomposition Approach to Multiregional Environmental Planning: A Numerical Study. In C. Carraro and A. Haurie. editors. *Operations Research and Environmental Management.* pages 119 - 132. Kluwer Academic Publishers. 1996.

[4] R.R. Barton. D.W. Hearn. and S. Lawphongpanich. The Equivalence of Transfer and Generalized Benders Decomposition Methods for Traffic Assignment. *Transportation Research B.* 23B(1):61–73. 1989.

[5] M. S. Bazaraa. *Linear Programming and Network Flows.* John Wiley and Sons Inc.. 2 edition. 1990.

[6] A. Brooke. D. Kendrick. and A. Meeraus. *GAMS: A User's Guide. Release 2.25.* The Scientific Press. 1992.

[7] Benno Bueler. Solving an Equilibrium Model for Trade of CO_2 Emission Permits. *European Journal of Operational Research.* 102:393–403. 1997.

[8] W. Chung. Y. June Wu. and J. David Fuller. Dynamic Energy and Environment Equilibrium Model for the Assessment of CO_2 Emission Control in Canada and the USA. *Energy Economics.* 19:103–124. 1997.

[9] G.B. Dantzig and P. Wolfe. The Decomposition Algorithm for Linear Programs. *Econometrica.* 29(4):767–778. 1961.

[10] Y.M.I. Dirickx and L.P. Jennergren. *Systems Analysis by Multilevel Methods*. Wiley IIASA International Series on Applied Systems Analysis. Pitman Press. 1979.

[11] J.D. Fuller and D. Luthra. Model Formulation with the Waterloo Energy Modelling System (WATEMS). *Energy*. 15(5):413–425. 1990.

[12] J.D. Fuller and Y. Wu. Effects of Supply Elasticity on Convergence of the PIES Algorithm. *Operations Research Letters*. 8:271–227. 1989.

[13] P.T. Harker and J.S. Pang. Finite-dimensional Variational Inequality and Nonlinear Complementarity Problems: A Survey of Theory. Algorithms and Applications. *Mathematical Programming*. 48:161–220. 1990.

[14] D. W. Hearn. Practical and Theoretical Aspects of Aggregation Problems in Transportation Planning Models. In *Transportation Planning Models*. Elsevier Science Publishers. 1984.

[15] B. Lan and J.D. Fuller. A Primal-Dual Nested Decomposition Algorithm for Multistage Linear Programs. Technical report. Department of Management Sciences. University of Waterloo. Waterloo. Ontario. Canada N2L 3G1. 1996.

[16] S. Lawphongpanich and D.W. Hearn. Simplicial Decomposition of the Asymmetric Traffic Assignment Problem. *Transportation Research B*. 18B(2):123–133. 1984.

[17] S. Lawphongpanich and D.W. Hearn. Benders Decomposition for Variational Inequalities. *Mathematical Programming*. 48:231–247. 1990.

[18] R. Loulou and A. Kanudia. The Kyoto Protocol. Inter-Provincial Cooperation. and Energy Trading: A Systems Analysis with Integrated MARKAL Models. Technical Report G-98-42. GERAD and McGill University. Montreal: Canada. 1998.

[19] A. Mansur and J. Whalley. A Decomposition Algorithm for General Equilibrium Computation with Application to International Trade Models. *Econometrica*. 50(6):1547 - 1557. 1982.

[20] L. Mathiesen. Computational Experience in Solving Equilibrium Models by a Sequence of Linear Complementarity Problems. *Operations Research*. 33(6):1225–1250. 1985.

[21] F. H. Murphy. Making Large-Scale Models Manageable: Modeling from an Operations Management Perspective. *Operations Research*. 41(2):241 - 252. 1993.

[22] F. H. Murphy and M. V. Mudrageda. A Decomposition Approach for a Class of Economic Equilibrium Models. *Operations Research*. 46(3):368 - 377. 1998.

[23] F.H. Murphy. J.J. Conti. S.H. Shaw. and R. Snaders. Modeling and Forecasting Energy Markets with the Intermediate Future Forecasting System. *Operations Research*. 36(3):406–420. 1988.

[24] A. Nagurney. *Network Economics: A Variational Inequality Approach*. Kluwer Academic Publishers. 1993.

[25] E. Nurminski and T. Balabanov. Decomposition of a Large-scale Energy Model. *Large Scale System*. 4:295–308. 1983.

[26] J.S. Pang and C.S. Yu. Linearized Simplicial Decomposition Methods for Computing Traffic Equilibria on Networks. *Networks*. 14:427–438. 1984.

[27] J. Park. Parallel Primal-Dual Decomposition Method for Multi-Part Linear Progrmms. Technical report. Department of Management Sciences. University of Waterloo. Waterloo. Ontario. Canada N2L 3G1. 1998.

[28] M. Patriksson. *The Traffic Assignment Problem: Models and Methods*. The Netherlands:VSP. 1994.

[29] R.T. Rockafellar. *Convex Analysis*. Princeton University Press. 1969.

[30] H. E. Scarf. The Approximation of Fixed Points of a Continuous Mapping. *SIAM Journal on Applied Mathematics*. 15:1328 - 1342. 1967.

[31] H. E. Scarf and T. Hansen. *Computation of Economic Equilibria*. Yale University Press. New Haven. 1973.

[32] A.H.K. Sham. On Convergence of the Decoupling Algorithm and Development of its Different Versions. Master's thesis. Department of Management Sciences. University of Waterloo. Waterloo. Ontario. Canada. 1997.

[33] G.L. Thompson and S. Thore. *Computational Economics: Economic Modelling with Optimization Software*. The Scientific Press. 1992.

[34] M.H. Wagner. Supply-Demand Decomposition of the National Coal Model. *Operations Research*. 29(6):1137–1153. 1981.

[35] Y. June Wu and William Chung. Assessing the Control of Energy-related CO_2 Emissions with a Dynamic Energy Process Model. *Energy. the International Journal*. 22(7):693–704. 1997.

[36] Y. June Wu and J. David Fuller. Introduction of Geometric. Distributed Lag Demand into Energy-Process Models. *Energy. the International Journal*. 20(7):647–656. 1995.

[37] Y. June Wu and J. David Fuller. An Algorithm for the Multiperiod Market Equilibrium Model with Geometric Distributed Lag Demand. *Operations Research*. 44(6):1002 - 1012. 1996.